Note

Before using this information and the product it supports, read the information in "Notices" on page 245.
## Appendix B. Support information

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using IBM Support Assistant</td>
<td>237</td>
</tr>
<tr>
<td>Tivoli Workload Scheduler IBM Support Assistant plug-in version and upgrade issues.</td>
<td>237</td>
</tr>
<tr>
<td>Searching knowledge bases</td>
<td>238</td>
</tr>
<tr>
<td>Searching the local information center</td>
<td>238</td>
</tr>
<tr>
<td>Obtaining fixes</td>
<td>239</td>
</tr>
<tr>
<td>Receiving weekly support updates</td>
<td>239</td>
</tr>
<tr>
<td>Contacting IBM Software Support</td>
<td>241</td>
</tr>
<tr>
<td>Determining the business impact</td>
<td>242</td>
</tr>
<tr>
<td>Describing problems and gathering information</td>
<td>242</td>
</tr>
<tr>
<td>Submitting problems</td>
<td>242</td>
</tr>
</tbody>
</table>

## Notices

For the code entitled Libmsg                                             | 246  |
For the code entitled Btree Library                                     | 247  |
For the code entitled ANT                                              | 247  |
For the code entitled Regex                                            | 247  |
For the code entitled SunRPC                                            | 248  |
For System Service Monitors 4.0 Code                                    | 248  |
Libpcap                                                               | 248  |
Net SNMP                                                               | 248  |
BZIP2                                                                 | 250  |
MD5                                                                    | 250  |
INSTALLSHIELD MULTI-PLATFORM 11.5 Code                                  | 251  |
APACHE CODE                                                            | 251  |
JXML CODE                                                              | 252  |
InfoZip Code                                                           | 253  |
HSQL CODE                                                              | 254  |
BROWSERLAUNCHER code                                                   | 255  |
JTOPEN Code                                                            | 255  |
For the code entitled embedded version of IBM WebSphere Application Server (“embedded WebSphere Application Server”) V6.1 | 255  |
Apache code                                                            | 256  |
CUP Parser Generator for Java                                          | 259  |
EPL CODE                                                               | 260  |
JYTHON                                                                 | 260  |
XML SCHEMAS                                                            | 261  |
ICU v 3.4 International Components for Unicode (ICU):                 | 262  |
OASIS XML SCHEMAS                                                      | 262  |
For the code entitled Time Zone Library                                 | 263  |
For the code entitled SNMP Library                                     | 263  |
For the code entitled TCL                                              | 264  |
## List of figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Single domain topology</td>
<td>6</td>
</tr>
<tr>
<td>2.</td>
<td>Internetwork dependencies</td>
<td>7</td>
</tr>
<tr>
<td>3.</td>
<td>Multiple domain topology</td>
<td>8</td>
</tr>
<tr>
<td>4.</td>
<td>Multiple inbound connections architecture</td>
<td>10</td>
</tr>
<tr>
<td>5.</td>
<td>DB2 client database connection</td>
<td>14</td>
</tr>
<tr>
<td>6.</td>
<td>Event Generation Flow</td>
<td>205</td>
</tr>
<tr>
<td>7.</td>
<td>Tivoli Workload Scheduler situations</td>
<td>218</td>
</tr>
<tr>
<td>8.</td>
<td>Situation content</td>
<td>219</td>
</tr>
<tr>
<td>9.</td>
<td>IBM Tivoli Monitoring architecture</td>
<td>223</td>
</tr>
</tbody>
</table>
# List of tables

1. Command syntax ........................................... xxii
2. Tivoli Workload Scheduler installation scenarios .................. 13
3. Temporary directories required size ............................. 16
4. Required authorization roles for running the installation wizard .................. 16
5. Required authorization roles for running the installation wizard .................. 17
6. Required authorization roles for running twsinst .................. 17
7. Required authorization roles for Software Distribution ............... 17
8. Installation element validation criteria .................. 20
9. Symlink options ........................................... 23
10. Upgrade availability for Tivoli Workload Scheduler components ............. 24
11. Engine supported platforms .................................. 27
12. Minimum required versions of DB2 ................................ 29
13. Software Requirements for Tivoli Workload Scheduler .................. 30
14. Software requirements for the bundled DB2 .................. 33
15. Disk space requirements, in megabytes, for Tivoli Workload Scheduler components and features .................. 37
16. Temporary file space, in megabytes, required by the installation .................. 37
17. Memory requirements, in megabytes ................................ 38
18. Database installation flow selection ................................ 44
19. Optional installable features and components .................. 58
20. Response files ........................................... 60
21. SPB installation parameters .................................. 64
22. List of parameters to install language packs .................. 67
23. Objects downloaded by the tutorial utility .................. 70
24. List of scheduling scenarios .................................. 71
25. Data and behavior mapping for domains .................. 105
26. Data and behavior mapping for workstations .................. 105
27. Data and behavior mapping for workstation classes .................. 106
28. Data and behavior mapping for job definitions .................. 107
29. Data and behavior mapping for Windows users .................. 107
30. Data and behavior mapping for resources .................. 107
31. Data and behavior mapping for job streams .................. 107
32. Data and behavior mapping for jobs .................. 108
33. Data and behavior mapping for dependencies .................. 109
34. Shortcuts for encryption ciphers .................. 157
35. Files for Local Options .................................. 169
36. Type of communication depending on the security level value .................. 169
37. Tivoli Workload Scheduler/NetView events .................. 184
38. Events filtered by CHSCHED .................. 187
39. Enterprise-specific traps .................................. 190
40. Tivoli Workload Scheduler engine event formats .................. 197
41. Positional variables for events 101-118,120-122,204 (job events) .................. 198
42. Positional variables for event 119 (job property modified) .................. 199
43. Positional variables for events 151-161,163-165 (schedule events) .................. 199
44. Positional variables for event 162 (schedule property modified) .................. 200
45. Positional variables for event 202 (schedule prompt) .................. 200
46. Positional variables for event 203 (job prompt) .................. 200
47. CHSCHED event filtered .................. 202
48. Job scheduling events .................. 206
49. Events formats table .................. 213
50. Events formats table .................. 219
51. Events formats table .................. 220
52. Events formats table .................. 220
53. Events formats table .................. 221
54. Events formats table .................. 221
55. Events formats table .................. 221
56. Events formats table .................. 221
57. Events formats table .................. 222
58. CHSCHED filtered events .................. 227
59. Tivoli Workload Scheduler engine event formats for Tivoli Enterprise Portal .................. 228
60. Positional variables for events 101-118,120-122,204 (job events) .................. 230
61. Positional variables for events 151-161,163-165 (job stream events) .................. 231
About this publication

This IBM® Tivoli® Workload Scheduler Planning and Installation Guide provides information for planning, installing, migrating, and configuring an IBM Tivoli Workload Scheduler network.

What is new in this release

For information about the new or changed functions in this release, see Tivoli Workload Scheduling Suite: Overview.

For information about the APARs that this release addresses, see the Tivoli Workload Scheduler Download Document at [http://www.ibm.com/support/docview.wss?rs=672&uid=swg24016672](http://www.ibm.com/support/docview.wss?rs=672&uid=swg24016672).

What is new in this publication

This section describes what has changed in this publication since version 8.3 Fix Pack 01.

Note: Changed or added text is marked by a revision bar in the left margin.

For version 8.4, the following changes were made:

Part 1. Preparing for installation

- Chapter 1, “Getting started,” on page 3 has been updated with the following information:
  - Support information for Internet Protocol version 6 (IPv6) was added in “Support for Internet Protocol version 6” on page 12
  - Information about RDBMS (specifically DB2® version 9.1) concerning Tivoli Workload Scheduler installation was documented in “Supported Relational Data Base Management Systems” on page 12
  - An overview paragraph about the event-driven workload automation function was added in “Event-driven workload automation” on page 11
  - “The installation CDs” on page 21 section has been updated to reflect the new CDs in this version.

- Chapter 2, “Supported operating systems and requirements,” on page 27 has been added to describe supported operating systems and prerequisites.

Part 2. Installing, upgrading and uninstalling

- Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41 has been updated to add the Tivoli Workload Scheduler report data table spaces and new port numbers needed by the Embedded Version of the WebSphere Application Server version 6.1.

- Chapter 4, “Using the Tivoli Workload Scheduler tutorial utility,” on page 69 has been added to describe the Tivoli Workload Scheduler tutorial utility.

- Three new upgrade chapters have been added to describe the upgrade procedures from previous versions.
What is new

- Chapter 5, “Upgrading a version 8.2 or 8.2.1 master domain manager instance,” on page 77
- Chapter 6, “Upgrading a version 8.3 master domain manager instance,” on page 111
- Chapter 7, “Upgrading agents,” on page 125

Part 3. Configuring

- Chapter 10, “Performing optional customization,” on page 143 has been updated to include new global and local options for event rule management and Websphere Application Server support in “Setting global options” on page 143, and “Setting local options” on page 150.

Appendixes

- Three new integrations with the following products have been added to Appendix A, “Integration with other IBM Tivoli products,” on page 179
  - “Integration with Tivoli Monitoring” on page 217, containing two sections: “Integration with IBM Tivoli Monitoring using a script data provider” on page 217, and “Integration with Tivoli Enterprise Portal using a file data provider” on page 222
  - “Integration with Tivoli Storage Manager” on page 232
  - “Integration with Tivoli License Compliance Manager” on page 235

Who should read this publication

This guide is intended for the following audience:

- Tivoli Workload Scheduler administrators - those who plan the network topology
- Installers - those who install the network

What this publication contains

This guide contains the following parts, chapters, and appendixes:

Part 1. Preparing for installation contains the following chapters:

- Chapter 1, “Getting started,” on page 3
  Describes what you need to know to plan your network and the features available.
- Chapter 2, “Supported operating systems and requirements,” on page 27
  Describes the supported operating systems and requirements for the installation.

Part 2. Installing, upgrading, and uninstalling contains the following chapters:

- Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41
  Describes installation using the various installation methods.
- Chapter 4, “Using the Tivoli Workload Scheduler tutorial utility,” on page 69
  Describes how to install and use the Tivoli Workload Scheduler tutorial.
- Chapter 5, “Upgrading a version 8.2 or 8.2.1 master domain manager instance,” on page 77
  Describes upgrading from versions 8.2 and 8.2.1.
- Chapter 6, “Upgrading a version 8.3 master domain manager instance,” on page 111
  Describes upgrading from version 8.3.
- Chapter 7, “Upgrading agents,” on page 125
Describes upgrading agents from previous versions.

- **Chapter 8, “Uninstalling Tivoli Workload Scheduler,” on page 133** Describes how to uninstall Tivoli Workload Scheduler.

**Part 3. Configuring** contains the following chapters:

- **Chapter 9, “Configuring after installation,” on page 139** Describes the configuration you need to do when you have finished installing.
- **Chapter 10, “Performing optional customization,” on page 143** Describes tuning of your installation.
- **Chapter 11, “Setting connection security,” on page 165** Describes how you set your security parameters.

**Part 4. Appendixes** contains the following appendixes:

- **Appendix A, “Integration with other IBM Tivoli products,” on page 179** Describes what you need to know to integrate Tivoli Workload Scheduler with other Tivoli products.
- **Appendix B, “Support information,” on page 237** Describes all the information you need to obtain support from IBM.

---

**Publications**

This section lists publications in the *Tivoli Workload Scheduler* library and any other related publications. It also describes how to access Tivoli publications online and how to order Tivoli publications.

**Tivoli Workload Scheduler library**

Tivoli Workload Scheduler comprises several separate products available for a variety of operating systems. The library is divided into:

- **IBM Tivoli Workload Scheduling suite library**
  This library contains all cross-platform and cross-product publications for Tivoli Workload Scheduler.

- **IBM Tivoli Workload Scheduler distributed library**
  This library contains all of the publications that refer to the use of Tivoli Workload Scheduler on operating systems other than z/OS®.

- **IBM Tivoli Workload Scheduler for z/OS library**
  This library contains all publications that apply only to IBM Tivoli Workload Scheduler for z/OS.

- **IBM Tivoli Workload Scheduler for Applications library**
  This library contains all publications that apply only to IBM Tivoli Workload Scheduler for Applications.

- **IBM Tivoli Workload Scheduler for Virtualized Data Centers library**
  This library contains all publications that apply only to IBM Tivoli Workload Scheduler for Virtualized Data Centers.
Publications

IBM Tivoli Workload Scheduling suite library
The following publications are available in the IBM Tivoli Workload Scheduling suite library. This library includes publications that are common to all products, operating systems, and components.

- **Tivoli Workload Scheduling Suite: Overview, SC32-1256**
  Provides an overview of all Tivoli Workload Scheduler products, and the way they can be used together to provide workload management solutions for your whole enterprise.

  **Note:** This manual used to be called "General Information".

- **Tivoli Workload Scheduler: Dynamic Workload Console Installation and Troubleshooting Guide, SC32-1572**
  Describes how to work with Tivoli Workload Scheduler, regardless of operating system, using a common Web-based GUI called the Dynamic Workload Console.

  This provides full information about downloading the product CD images. It also indicates the APARs that have been fixed in this release.

  This provides full information about the hardware and software prerequisites of the product.

- **Tivoli Workload Scheduler: Job Scheduling Console User’s Guide, SC32-1257**
  Describes how to work with Tivoli Workload Scheduler, regardless of operating system, using a common GUI called the Job Scheduling Console.

- **Tivoli Workload Scheduler, version 8.3: Warehouse Enablement Pack version 1.1.1 Implementation Guide for Tivoli Data Warehouse, versions 1.2 and 1.3**
  Provides information about enabling Tivoli Workload Scheduler for Tivoli Data Warehouse. This publication is only available on the Tivoli Workload Scheduler Engine Installation CD "TWS84_OTHER", in the following path:
  `TDW_enablement_pack/tdw_weps/aws/v8300/doc/itws_for_TDW.doc`
  You cannot access it online, in the same way that you can the other books (see "Accessing publications online" on page xix).

IBM Tivoli Workload Scheduler distributed library
The following publications are available in the IBM Tivoli Workload Scheduler distributed library. This library contains publications that refer to using the product on distributed operating systems (all operating systems except z/OS).

- **Tivoli Workload Scheduler: Quick Start Guide, GC23-6141**
  Provides information on how to get started with an installation of Tivoli Workload Scheduler on distributed operating systems.

  This provides full information about downloading the product CD images. It also indicates the APARs that have been fixed in this release.

  This provides full information about the hardware and software prerequisites of the product.

- **Tivoli Workload Scheduler: Planning and Installation Guide, SC32-1273**
Describes how to plan for and install IBM Tivoli Workload Scheduler on distributed operating systems, and how to integrate Tivoli Workload Scheduler with NetView®, Tivoli Data Warehouse, Tivoli Monitoring, and Tivoli Enterprise Console®.

- **Tivoli Workload Scheduler: Reference Guide, SC32-1274**
  Provides an explanation of the concepts of Tivoli Workload Scheduler and describes how the product is used. Also describes the Tivoli Workload Scheduler command line used on distributed operating systems, and how extended and network agents work.

- **Tivoli Workload Scheduler: Administration and Troubleshooting, SC32-1275**
  Provides information about how to administer Tivoli Workload Scheduler on distributed operating systems, and what to do if things go wrong. It includes help on many messages generated by the main components of Tivoli Workload Scheduler.

- **Tivoli Workload Scheduler: Database Views, SC32-2261**
  Provides information about the views of the IBM Tivoli Workload Scheduler database.

- **Tivoli Workload Scheduler: Using Microsoft Cluster Service on Windows Server 2003, SC23-6119**
  Describes how to use Tivoli Workload Scheduler with the Microsoft® Cluster service on Windows® Server 2003 to achieve high availability.

- **Tivoli Workload Scheduler: Limited Fault-tolerant Agent for i5/OS, SC32-1280**
  Describes how to install, configure, and use Tivoli Workload Scheduler limited fault-tolerant agents on i5/OS.

- **Java™ API documentation.**
  Provides information about using the Java Application Programming Interface (API). This is a set of available classes and methods running in a JAVA environment that you use to create a custom interface to manage scheduling objects in the database and in the plan. They cannot be used to manage the plan or to set global options.

  Documentation for the API is provided on all distributed product CDs. Mount the CD for your platform and open the following file with your Internet browser: `<CD_drive>/API/doc/index.html`.


**IBM Tivoli Workload Scheduler for z/OS library**

The following publications are available in the Tivoli Workload Scheduler for z/OS library:

- **Tivoli Workload Scheduler for z/OS: Getting Started, SC32-1262**
  Discusses how to define your installation data for Tivoli Workload Scheduler for z/OS and how to create and modify plans.

- **Tivoli Workload Scheduler for z/OS: Installation Guide, SC32-1264**
  Describes how to install Tivoli Workload Scheduler for z/OS.

- **Tivoli Workload Scheduler for z/OS: Customization and Tuning, SC32-1265**
  Describes how to customize Tivoli Workload Scheduler for z/OS.
Publications

- **Tivoli Workload Scheduler for z/OS: Managing the Workload, SC32-1263**
  Explains how to plan and schedule the workload and how to control and monitor the current plan.

- **Tivoli Workload Scheduler for z/OS: Quick Reference, SC32-1268**
  Provides a quick and easy consultation reference to operate Tivoli Workload Scheduler for z/OS.

- **Tivoli Workload Scheduler for z/OS: Diagnosis Guide and Reference, SC32-1261**
  Provides information to help diagnose and correct possible problems when using Tivoli Workload Scheduler for z/OS.

- **Tivoli Workload Scheduler for z/OS: Messages and Codes, SC32-1267**
  Explains messages and codes in Tivoli Workload Scheduler for z/OS.

- **Tivoli Workload Scheduler for z/OS: Programming Interfaces, SC32-1266**
  Provides information to write application programs for Tivoli Workload Scheduler for z/OS.

- **Tivoli Workload Scheduler for z/OS: Scheduling End-to-end, SC32-1732**
  Provides information on how to integrate Tivoli Workload Scheduler for z/OS with Tivoli Workload Scheduler, controlling workload in a distributed environment from a z/OS master domain manager.

- **Tivoli Workload Scheduler for z/OS: Memo to Users, GI11-4209**
  Provides a summary of changes for the current release of the product.

- **Tivoli Workload Scheduler for z/OS: Program Directory, GI11-4248**
  Provided with the installation tape for Tivoli Workload Scheduler for z/OS, describes all of the installation materials and gives installation instructions specific to the product release level or feature number.


**IBM Tivoli Workload Scheduler for Applications library**

The following publications are available in the IBM Tivoli Workload Scheduler for Applications library:

- **Tivoli Workload Scheduler for Applications: User’s Guide, SC32-1278**
  Provides information on how to install, set up, and use the IBM Tivoli Workload Scheduler access methods that run and control jobs of the following applications:
  - Oracle
  - PeopleSoft
  - R/3
  - z/OS

- **Tivoli Workload Scheduler for Applications: Quick Start Guide, GC32-1538**
  Gives an overview on how to get started with Tivoli Workload Scheduler for Applications.

  This provides full information about downloading the product CD images. It also indicates the APARs that have been fixed in this release.

  This provides full information about the hardware and software prerequisites of the product.

**IBM Tivoli Workload Scheduler for Virtualized Data Centers library**

The following publications are available in the IBM Tivoli Workload Scheduler for Virtualized Data Centers library:

- **Tivoli Workload Scheduler for Virtualized Data Centers: User’s Guide, SC32-1454**
  
  Describes how to extend the scheduling capabilities of Tivoli Workload Scheduler to workload optimization and grid computing by enabling the control of IBM LoadLeveler® and IBM Grid Toolbox jobs.

- **Tivoli Workload Scheduler for Virtualized Data Centers: Release Notes, SC32-1453**
  
  Provides late-breaking information about Tivoli Workload Scheduler for Virtualized Data Centers.


**Related publications**

The following documents provide additional information:

- **IBM Redbooks: Getting Started with IBM Tivoli Workload Scheduler V8.3: Best Practices and Performance Improvements, SG24-7237**

  Abstract: IBM Tivoli Workload Scheduler is an IBM strategic scheduling product that runs on different platforms including the mainframe. The new version of the product, IBM Tivoli Workload Scheduler V8.3, comes with some important enhancements, such as relational database management system support, new advanced planning system, which allows the definition of plans that span more that 24 hours, removal of framework requirements, new application programming interface (API), Job Scheduling Console enhancements, and so on. This IBM Redbook documents the architecture, deployment, best practices, and migration scenarios for IBM Tivoli Workload Scheduler V8.3 in a distributed environment. In addition, it covers IBM Tivoli Workload Scheduler V8.3 security, IBM DB2 and IBM WebSphere considerations, troubleshooting, tuning for performance, application programming interface, and JnextPlan, which has replaced the JnextDay process in this release.

  Clients and Tivoli professionals who are responsible for installing, administering, maintaining, or using IBM Tivoli Workload Scheduler V8.3 will find this book a major reference.

  [This Redbook can be found on the Redbooks™ Web site at http://www.redbooks.ibm.com/abstracts/sg247237.html](http://www.redbooks.ibm.com/abstracts/sg247237.html)

- **IBM Redbooks: Customizing IBM Tivoli Workload Scheduler for z/OS V8.2 to Improve Performance, SG24-6352**

  Abstract: This IBM Redbook covers the techniques that can be used to improve the performance of Tivoli Workload Scheduler for z/OS (including end-to-end scheduling).

  [This Redbook can be found on the Redbooks Web site at http://www.redbooks.ibm.com/abstracts/sg246352.html](http://www.redbooks.ibm.com/abstracts/sg246352.html)

- **IBM Redbooks: IBM Tivoli Workload Scheduler for z/OS: Best Practices, SG24-7156**

  Abstract: This IBM Redbook describes best practices for using Tivoli Workload Scheduler for z/OS. Topics covered include:

  - Installation best practices
  - Installation verification
Publications

- Started tasks
- Communication
- Initialization statements and parameters
- Security
- Exits
- Restart and cleanup
- Dataset triggering and event trigger tracking
- Variables
- Audit report facility

This Redbook can be found on the Redbooks Web site at [http://www.redbooks.ibm.com/abstracts/sg247156.html](http://www.redbooks.ibm.com/abstracts/sg247156.html)

**• IBM Redbooks: Integrating IBM Tivoli Workload Scheduler with Tivoli Products, SG24-6648**

**Abstract:** This IBM Redbook explains the benefits and technical merits of integrating IBM Tivoli Workload Scheduler Distributed and IBM Tivoli Workload Scheduler for z/OS with other IBM products. Scheduling is a mission critical process for any company. However, when you talk about scheduling, you are really talking about an ecosystem. In this ecosystem, each solution is a building block that adds value to the overall solution. With IBM Tivoli Workload Scheduler, you can collect and add data to and from each component. In addition, expanding the scheduling ecosystem to include monitoring, management, help desk, storage, and business systems management provides greater value.

This book discusses all these integration points and provides detailed scenarios on how to integrate IBM Tivoli Workload Scheduler with these types of applications. Because workload management is widely considered the nucleus of the data center, there are numerous opportunities for you to integrate IBM Tivoli Workload Scheduler with other products. This book addresses just some of these many opportunities. In terms of integration with IBM Tivoli Workload Scheduler, do not limit yourself to the products that this book discusses. Integration points discussed in this book should give you an idea of the potential value that IBM Tivoli Workload Scheduler integration can provide for your company.

This Redbook can be found on the Redbooks Web site at [http://www.redbooks.ibm.com/abstracts/sg246648.html](http://www.redbooks.ibm.com/abstracts/sg246648.html)

**• IBM Redbooks: WebSphere Application Server V6 System Management & Configuration Handbook, SG24-6451**

**Abstract:** This IBM Redbook provides system administrators, developers, and architects with the knowledge to configure a WebSphere Application Server V6 runtime environment, to package and deploy Web applications, and to perform ongoing management of the WebSphere environment.

One in a series of handbooks, the entire series is designed to give you in-depth information about the entire range of WebSphere Application Server V6 runtime environments and administration process.

This Redbook can be found on the Redbooks Web site at [http://www.redbooks.ibm.com/abstracts/sg246451.html](http://www.redbooks.ibm.com/abstracts/sg246451.html)
Accessing terminology online

The Tivoli Software Glossary includes definitions for many of the technical terms related to Tivoli software. The Tivoli Software Glossary is available at the following Tivoli software library Web site:

http://publib.boulder.ibm.com/tividd/glossary/tivoliglossarymst.htm

The IBM Terminology Web site consolidates the terminology from IBM product libraries in one convenient location. You can access the Terminology Web site at the following Web address:

http://www.ibm.com/software/globalization/terminology

Accessing publications online

The Tivoli Workload Scheduler documentation CD contains the publications that are in the product library. The format of the publications is PDF, HTML, or both. The publications are found within a Tivoli Information Center. Place the CD in the CD drive of a Windows computer and the Information Center automatically opens. If the Information Center does not open automatically, or you require more information, consult the readme.txt file in the root of the CD.

IBM posts publications for this and all other Tivoli products, as they become available and whenever they are updated, to the Tivoli software information center Web site. There are two ways of accessing the Tivoli software information center:

Directly access the IBM Tivoli Workload Scheduler Information Center

Go directly to the Information Center at the following Web address:


Access the IBM Tivoli Workload Scheduler Information Center from the Tivoli Technical Product Documents Web site

Access the Tivoli software information center by following these steps:

1. Go to the Tivoli library at the following Web address:


2. Click Tivoli product manuals

3. In the Tivoli Technical Product Documents Alphabetical Listing window, click W (for Workload Scheduler) or scroll down to the W section of the product list

4. Click the appropriate Tivoli Workload Scheduler product link to access your product libraries at the Tivoli software information center. All publications in the Tivoli Workload Scheduler suite library, distributed library, and z/OS library can be found under the entry Tivoli Workload Scheduler. The Tivoli software information center page for Tivoli Workload Scheduler is displayed. It gives you access to the publications relating to the latest version of the product. Links are provided to the documentation of prior versions.

5. Click to access the Tivoli Workload Scheduler Information Center. The Information Center is Eclipse-based, and contains full instructions on how to use it to obtain information and search the publications for specific terms.
Note: If you print PDF publications on other than letter-sized paper, set the option in the File → Print window that enables Adobe Reader to print letter-sized pages on your local paper.

For all types of information about DB2, go to the DB2 Information Center: http://publib.boulder.ibm.com/infocenter/db2luw/v8//index.jsp

For all types of information about the Embedded Version of the WebSphere Application Server version 6.1, go to the WebSphere® Application Server Information Center: http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp

Note: The Embedded Version of the WebSphere Application Server version 6.1 is not the same as Websphere Application Server - Express. It is a runtime version of Websphere Application Server, version 6.1 which is bundled in and managed by Tivoli Workload Scheduler.

For all types of information about the Oracle database, consult the documentation of Oracle Corporation. When this manual was published, the relevant documentation could be found on http://www.oracle.com/technology/documentation/index.html

Note: This information has been included as a courtesy, and IBM cannot guarantee that this URL will continue to be correct.

Tivoli Workload Scheduler online books
All the books in the Tivoli Workload Scheduler for z/OS library are available in displayable softcopy form on CD in the IBM Online Library: z/OS Software Products Collection Kit, SK3T-4270. You can read the softcopy books on CD using these IBM licensed programs:

- BookManager® READ/2 (program number 5601-454)
- BookManager READ/DOS (program number 5601-453)
- BookManager READ/6000 (program number 5765-086)

All the BookManager programs need a personal computer equipped with a CD drive (capable of reading disks formatted in the ISO 9660 standard) and a matching adapter and cable. For additional hardware and software information, refer to the publications for the specific BookManager product you are using.

Updates to books between releases are provided in softcopy only.

Ordering publications
You can order many Tivoli publications online at the following Web site: http://www.elink.ibmlink.ibm.com/public/applications/publications/cgibin/pbi.cgi

You can also order by telephone by calling one of these numbers:
- In the United States: 800-879-2755
- In Canada: 800-426-4968
- In other countries, contact your software account representative to order Tivoli publications. To locate the telephone number of your local representative perform the following steps:
  1. Go to the following Web site:
2. Select your country from the list and click . The IBM Publications Center page is displayed.
3. Click About this site in the main panel to see an information page which includes the telephone number of your local representative.

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

For additional information, see the Accessibility Appendix in the Tivoli Workload Scheduler Job Scheduling Console User’s Guide.

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:
• Searching knowledge bases: You can search across a large collection of known problems and workarounds, Technotes, and other information.
• Obtaining fixes: You can locate the latest fixes that are already available for your product.
• Contacting IBM Software Support: If you still cannot solve your problem, and you need to work with someone from IBM, you can use a variety of ways to contact IBM Software Support.

For more information about these three ways of resolving problems, see Appendix B, “Support information,” on page 237.

Conventions used in this publication

This publication uses several conventions for special terms and actions, operating system-dependent commands and paths, command syntax, and margin graphics.

Typeface conventions

This publication uses the following typeface conventions:

Bold
• Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
• Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as Tip, and Operating system considerations)
• Keywords and parameters in text
Conventions

Italic

- Words defined in text
- Emphasis of words (words as words)
- New terms in text (except in a definition list)
- Variables and values you must provide

Monospace

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

Operating system-dependent variables and paths

This publication uses the UNIX® convention for specifying environment variables and directory notation, except where the context or the example path is specifically Windows.

When using the Windows command line, replace $\text{variable}$ with %variable% for environment variables and replace each forward slash (/) with a backslash (\) in directory paths. The names of environment variables are not always the same in Windows and UNIX environments. For example, %TEMP% in Windows is equivalent to $\text{tmp}$ in UNIX environments.

Note: If you are using the bash shell on a Windows system, you can use the UNIX conventions.

Command syntax

This publication uses the following syntax wherever it describes commands:

Table 1. Command syntax

<table>
<thead>
<tr>
<th>Syntax convention</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name of command</td>
<td>The first word or set of consecutive characters.</td>
<td>conman</td>
</tr>
<tr>
<td>Brackets ([ ] )</td>
<td>The information enclosed in brackets ([ ]) is optional. Anything not enclosed in brackets must be specified.</td>
<td>[-file definition_file]</td>
</tr>
<tr>
<td>Braces ( { } )</td>
<td>Braces ( { } ) identify a set of mutually exclusive options, when one option is required.</td>
<td>[-prompts</td>
</tr>
<tr>
<td>Underscore (_)</td>
<td>An underscore (_) connects multiple words in a variable.</td>
<td>prompt_name</td>
</tr>
<tr>
<td>Vertical bar (</td>
<td>)</td>
<td>Mutually exclusive options are separated by a vertical bar (</td>
</tr>
<tr>
<td>Syntax convention</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Bold</strong></td>
<td><strong>Bold</strong> text designates literal information that must be entered on the command line exactly as shown. This applies to command names and non-variable options.</td>
<td><strong>composer add</strong> file_name</td>
</tr>
<tr>
<td><strong>Italic</strong></td>
<td>Italic text is variable and must be replaced by whatever it represents. In the example to the right, the user would replace file_name with the name of the specific file.</td>
<td>file_name</td>
</tr>
</tbody>
</table>
| **Ellipsis (...)** | An ellipsis (...) indicates that the previous option can be repeated multiple times with different values. It can be used inside or outside of brackets. | \[-x \text{file}_\text{name}]...

An ellipsis outside the brackets indicates that \(-x\) file_name is optional and may be repeated as follows: \(-x\) file_name1 \(-x\) file_name2 \(-x\) file_name3 \(-x\) file_name...

An ellipsis inside the brackets indicates that \(-x\) file_name is optional, and the file variable can be repeated as follows: \(-x\) file_name1 file_name2 file_name3

\(-x\) file_name \(-x\) file_name...

An ellipsis used with this syntax indicates that you must specify \(-x\) file_name at least once.
Conventions
Part 1. Preparing for installation
Chapter 1. Getting started

This chapter gives you an overview of what you need to know to prepare for installation. It consists of the following sections:

- “Planning the network”
- “Planning installations” on page 12
- “Engine coexistence and upgrade notes” on page 24

Planning the network

This section provides information to help you plan your Tivoli Workload Scheduler network.

Network overview

This section describes the topology of a Tivoli Workload Scheduler network.

A Tivoli Workload Scheduler network consists of the workstations on which jobs and job streams are run. Primarily, workstation definitions refer to physical workstations. However, for extended agents the workstations are logical definitions that must be hosted by a physical workstation.

When you are installing the network, you can install engine instances with the following workstation types:

- Master Domain Manager (MDM)
- Backup Master
- Agent

Master Domain Manager and Backup Master workstations can also perform the following role for the event-driven workload automation feature:

- Event processing server

Agent workstations can also perform the following roles:

- Domain Manager
- Hosts for extended agents
- Standard Agent
- Fault-tolerant agent

As well as the various agent types, you can choose to install the following additional features:

- Command-line client
- Connector

Note: For a more detailed description of the Tivoli Workload Scheduler components, refer to Overview.
Tivoli Workload Scheduler network overview

A Tivoli Workload Scheduler network contains at least one domain, the master domain, in which the master domain manager is the management hub. Additional domains can be used to divide a widely distributed network into locally managed groups.

Using multiple domains reduces the amount of network traffic by reducing communication between the master domain manager and workstations. In a single domain configuration the master domain manager maintains communications with all of the workstations in the network.

In a multi-domain configuration the master domain manager communicates with the workstations in its domain and subordinate domain managers. The subordinate domain managers communicate with the workstations in their domains and subordinate domain managers. Multiple domains also provide fault-tolerance by limiting the problems caused by losing a domain manager in a single domain. To limit the effects further, you can designate backup domain managers to take over if their domain managers fail.

Tivoli Workload Scheduler can operate with the following services and protocols:

- Microsoft Active Directory
- Network Address Translation (NAT)
- Virtual Private Network (VPN)

Domains

When you define a new domain, you must identify the parent domain and the domain manager. The parent domain is the domain directly above the new domain in the domain hierarchy. All communications to and from a domain are routed through the parent domain manager.

Localized processing in your domain

Localized processing is separating your scheduling needs based on a common set of characteristics, such as geographical locations, business functions, and application groupings. Group related processing can limit the amount of interdependency information that needs to be communicated between domains.

The benefits of localized domains are:

- Decreased network traffic
  - Keeping processing localized to domains eliminates the need for frequent inter-domain communication.

- Tighter security and simplified administration
  - Security and administration can be defined at and limited to the domain level. Instead of network-wide or workstation-specific administration, you can have domain administration.

- Optimized network and workstation fault-tolerance
  - In a multiple domain network, you can define backups for each domain manager so that problems in one domain do not disrupt operations in other domains.
Network considerations

When you are planning your Tivoli Workload Scheduler network, you should consider the following:

- The number of workstations that comprise the network and the number of applications and jobs the network runs.
  The number of workstations in your network helps determine whether to use a single domain or a multiple domain architecture. If you have a small number of workstations, or a small number of applications to control, you do not need a multi-domain architecture.

- The number of geographic locations covered by your network and the reliability and efficiency of communication between the locations.
  Multiple geographic locations is one of the primary reasons for choosing a multiple domain architecture. One domain for each geographical location is a common configuration. A single domain architecture relies on the network maintaining continuous processing.

- The time zones of the geographic locations of your network.
  When your network is across multiple geographic locations, and those locations are in multiple time zones, you need to decide whether to activate the time zone feature. See "Time zone considerations" on page 11.

- Centralized or decentralized management.
  You can manage single or multiple domain networks from a single node, the master domain manager. If you want to manage multiple locations separately, you can consider the installation of a separate Tivoli Workload Scheduler network at each location. Some degree of decentralized management is possible in a standalone Tivoli Workload Scheduler network by mounting or sharing file systems.

- The type of applications that are run by your Tivoli Workload Scheduler.
  If you have multiple applications that are distinctly separate from other each other, you may choose to put them in separate domains.

- Whether you have a Windows network.
  When you have a Windows network, you might want your Tivoli Workload Scheduler domains to mirror your Windows domains.

- System performance and other criteria.
  You can define multiple domains to localize systems based on performance or operating system type.

- The amount of network traffic.
  If your network traffic is manageable, the need for multiple domains is less important.

- Dependencies between jobs.
  You need to plan for job dependencies that cross system boundaries, geographical boundaries, or application boundaries. For example, does the start of Job1 on workstation1 depend on the completion of Job2 running on workstation2. The degree of interdependence between jobs is an important consideration when planning your network. If you use multiple domains, you should try to keep interdependent objects in the same domain, thereby decreasing network traffic and improving use of the domain architecture. See the IBM Tivoli Workload Scheduler Job Scheduling Console User’s Guide.

- The level of fault-tolerance required.
A disadvantage of the single domain configuration is the reliance on a single domain manager. In a multi-domain network, the loss of a single domain manager affects only the agents in its domain.

- Firewalls.
  When your network contains firewalls you need to plan the structure of your domains around the firewalls. See “Working across firewalls” on page 177.

- Secure Sockets Layer (SSL) encryption.
  If you want to use SSL encryption in your network, you need to plan your domains in accordance with the protocol. See “Configuring the SSL connection protocol for the network” on page 170.

A single domain network
A single domain network consists of a master domain manager and any number of agents. Figure 1 shows an example of a single domain network. A single domain network is well suited to companies that have few locations and business functions. All communication in the network is routed through the master domain manager. With a single location, you are concerned only with the reliability of your local network and the amount of traffic it can handle.

![Diagram of a single domain network]

Figure 1. Single domain topology

Single domain networks can be combined with other networks, single or multiple domain, to meet multiple site requirements. Tivoli Workload Scheduler supports internetwork dependencies between jobs running on different networks.
The first example shows a single domain network. The master domain manager is located in Atlanta, along with several agents. There are also agents located in Denver. The agents in Denver depend on the master domain manager in Atlanta to resolve all interagent dependencies, even though the dependencies may be on jobs that run in Denver. An alternative would be to create separate single domain networks in Atlanta and Denver, as shown in the second example.

**A multiple domain network**

Multiple Domain networks are especially suited to companies that span multiple locations, departments, or business functions. A multiple domain network consists of a master domain manager, any number of lower tier domain managers, and any number of agents in each domain. Agents communicate only with their domain managers, and domain managers communicate with their parent domain managers.
As Figure 3 illustrates, the master domain manager is located in Atlantis. The master domain manager contains the database files used to document the scheduling objects, and distributes the Symphony file to its agents and the domain managers in Denver and Los Angeles. The Denver and Los Angeles domain managers then distribute the Symphony file to their agents and subordinate domain managers in Boulder, Aurora and Burbank. The master domain manager in Atlantis is responsible for broadcasting inter-domain information throughout the network.

All communication to and from the Boulder domain manager is routed through its parent domain manager in Denver. If there are schedules or jobs in the Boulder domain that are dependent on schedules or jobs in the Aurora domain, those dependencies are resolved by the Denver domain manager. Most interagent dependencies are handled locally by the lower tier domain managers, greatly reducing traffic on the network.
Backup master installation

When you install a backup master, the application server is installed within the Tivoli Workload Scheduler instance and is configured as for a master domain manager.

For a backup master installation, you supply the name of the Tivoli Workload Scheduler database used by the master during the installation. Installation checks for that the database exists and does not create it.

With your RDBMS (DB2 or Oracle) for the backup master, you can:

- Use an existing RDBMS server (Oracle only)
- Use an existing RDBMS remote server through an existing RDBMS local client
- Use an existing RDBMS remote server through a RDBMS local client installed during installation

It is not necessary to install a connector on a backup master in order to connect the Job Scheduling Console because the connector is installed automatically.

Switching to a backup domain manager

Each domain has a domain manager and, optionally, one or more backup domain managers. A backup domain manager must be in the same domain and be the same agent version as the domain manager it is backing up. The backup domain managers must be agents with resolve dependencies and full status options enabled in their workstation definitions.

If a domain manager fails during the production day, you can use the Job Scheduling Console or the switchmgr command in the command line, to switch to a backup domain manager. A switch manager action can be run by anyone with start and stop access to the domain manager and backup domain manager workstations.

A switch manager operation stops the backup manager, restarts it as the new domain manager, and converts the old domain manager to a fault-tolerant agent. The identities of the current domain managers are carried forward in the Symphony file from one processing day to the next, so any switch remains in effect until you switch back to the original domain manager.

Switch fault tolerance

The optional switch fault-tolerance mechanism is based on multiple inbound connections and switches roles between the domain manager and backup domain manager. In a domain, events are no longer routed from the primary domain manager, but arrive directly from the originating fault-tolerant agents. When a fault-tolerant agent sends an event to a primary domain manager, it also sends the event to all the full-status fault-tolerant agents in that domain. If it is unable to deliver the event to any of them the event is buffered in the corresponding pobox file on the fault-tolerant agent. It is preferable to limit the number of full status fault-tolerant agents to avoid an increase in network traffic.

Figure 4 on page 10 shows the multiple inbound connections architecture.
Network Overview

The solid arrows represent the connections that are created with a Tivoli Workload Scheduler without multiple inbound connections architecture. The dashed arrows represent the additional inbound connections that are created to the full-status fault-tolerant agent in a domain with multiple inbound connections architecture.

When the fault-tolerant switch is active, the link and unlink commands issued from the primary domain manager act both on the primary and on the secondary connections.

When a full-status fault-tolerant agent receives an event, it processes it and does not route it further, but buffers it locally in a cyclical queue called ftbox. ftbox acts as a recovery queue.

Since the impact of this reprocessing can be severe on performance, you should resize the ftbox files (using the evtsize command) to be large enough to hold the expected volume of events received from the moment the domain manager fails to the moment the switchmgr command is issued to recover from this failure.

The multiple inbound connections architecture ensures that all events received and processed by the primary domain manager are also received and processed by the full-status fault-tolerant agent (or are received or processed later if the events are still in a fault-tolerant agent pobox). If the primary domain manager fails, you can use the switch-manager command to switch the domain manager functionality from the primary domain manager to a selected full-status fault-tolerant agent.
When the switch-manager command is received, all the fault-tolerant agents in that domain disconnect from the primary domain manager and connect to the full-status fault-tolerant agent. During the link establishment phase, the new manager synchronizes with each connecting workstation by resending and regenerating the delta of the events that were buffered on the ftboxes, ensuring that none of the events still in the primary domain manager message boxes are lost or duplicated.

The full-status fault-tolerant agents are always updated with the latest status information and all the unprocessed or partially processed events are stored on at least two workstation (the original fault-tolerant agent and the domain manager, or the full-status fault-tolerant agent). The events can be resent and reprocessed, thus eliminating the single point of failure of the primary domain manager backup-domain manager communication.

**Note:** This approach applies both to top-down and bottom-up traffic. Inbound does not depend on the direction of the traffic, but is domain-centric, and is repeated the same way for each domain where at least one full-status fault-tolerant agent resides.

For information about configuring switch fault-tolerance, see “Setting global options” on page 143.

### Event-driven workload automation

Event-driven workload automation adds the capability to perform on-demand workload automation in addition to plan-based scheduling. The objective of event-driven workload automation in Tivoli Workload Scheduler is to carry out a predefined set of actions in response to events that occur on the nodes where Tivoli Workload Scheduler runs. This implies the capability to submit workload and run commands asynchronously, notify users via e-mail, or send messages to the Tivoli Enterprise Console.

The event-driven workload automation capability is based on event rules which are created and saved in the Tivoli Workload Scheduler database. On the agent nodes, a monitoring engine process captures the events occurring locally and sends them to the event processing server. The event processing server (typically the master domain manager or backup master domain manager) performs event correlation and runs the actions defined in the event rules. To configure and use the event-driven workload automation capability in your network, see the Reference Guide.

### Time zone considerations

Time zone support is an optional feature that is enabled by default. When enabled, time zone support allows you to manage workloads at a global level. For information on how to set the time zone, see “Setting global options” on page 143.

Time zone implementation also enables easy scheduling across multiple time zones.

For a description of how the time zone works, refer to *IBM Tivoli Workload Scheduler Reference Guide*. 

Network Overview
Network Overview

Workstation names

Job scheduling in a Tivoli Workload Scheduler network is distributed across multiple computers. To accurately track jobs, schedules, and other objects, each computer is given a unique workstation name. The names can be the same as network node names, as long as they comply with the naming rules of Tivoli Workload Scheduler. The maximum permitted length of a workstation name is sixteen bytes, composed of alphanumeric, dash (-), and underscore (_) characters. The name must start with a letter.

Planning installations

This section provides an overview of the prerequisites you need to check before you begin installing Tivoli Workload Scheduler. For a detailed list of supported operating systems and product prerequisites see Chapter 2, “Supported operating systems and requirements,” on page 27. It comprises the following topics:

- “Support for Internet Protocol version 6”
- “Embedded version of IBM WebSphere Application Server”
- “Supported Relational Data Base Management Systems”
- “Checking temporary directory requirements for installing using the installation wizard” on page 16
- “Checking user authorization requirements” on page 16
- “Tivoli Workload Scheduler user” on page 20
- “Product instances” on page 20
- “Installing for end-to-end scheduling” on page 20
- “Installation element criteria” on page 20
- “The installation CDs” on page 21
- “Installation paths” on page 22

Support for Internet Protocol version 6

Tivoli Workload Scheduler supports Internet Protocol version 6 (IPv6) in addition to the legacy IPv4. To help you in staging the transition from an IPv4 environment to a complete IPv6 environment, Tivoli Workload Scheduler provides IP dual-stack support. The product communicates using both IPv4 and IPv6 protocols simultaneously with other applications using IPv4 or IPv6.

Embedded version of IBM WebSphere Application Server

Connection between the Tivoli Workload Scheduler engine, the Job Scheduling Console, and the command line requires the Embedded Version of IBM Websphere Application Server 6.1. This is installed automatically when you install a master domain manager or backup master domain manager. In this document, it will be referred to as Websphere Application Server.

Supported Relational Data Base Management Systems

A Relational Data Base Management System (RDBMS) is required on master domain managers and on their backups. The RDBMS can be one of the following:

- DB2 Enterprise Server Edition
- Oracle

DB2 Server 9.1 is bundled with this version of the product. The main advantage of using DB2 9.1 is that it is enabled to work with IPv6 addresses. Fresh installations of Tivoli Workload Scheduler version 8.4 include the possibility to install DB2
Server 9.1 on computers that have no access to a RDBMS. However, the DB2 and Oracle Database versions supported by Tivoli Workload Scheduler version 8.3 are still supported and version 8.4 is designed to work with them if it finds them at installation time. The following matrix shows the possible scheduler-RDBMS version relationships:

Table 2. Tivoli Workload Scheduler installation scenarios

<table>
<thead>
<tr>
<th>Tivoli Workload Scheduler version 8.4 installation or upgrade scenario</th>
<th>RDBMS action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fresh install on computer with no existing RDBMS</td>
<td>DB2 9.1 is installed</td>
</tr>
<tr>
<td>Fresh install on computer with existing DB2 9.1</td>
<td>DB2 9.1 is acknowledged and accepted</td>
</tr>
<tr>
<td>Fresh install on computer with existing DB2 8.2</td>
<td>DB2 8.2 is acknowledged and accepted</td>
</tr>
<tr>
<td>Fresh install on computer with existing supported Oracle Database version</td>
<td>Oracle Database is acknowledged and accepted</td>
</tr>
<tr>
<td>Upgrade from version 8.3 with existing DB2 8.2</td>
<td>DB2 8.2 is acknowledged and accepted</td>
</tr>
<tr>
<td>Upgrade from version 8.3 with existing supported Oracle Database version</td>
<td>Oracle Database is acknowledged and accepted</td>
</tr>
</tbody>
</table>

The installation process (either fresh or upgrade from Version 8.3) does not provide the option to upgrade the DB2 version from 8.2 to 9.1. If you want to upgrade to DB2 9.1, you have to do so manually using the DB2 9.1 installation CDs packaged with Tivoli Workload Scheduler version 8.4. The CD also includes the DB2 9.1 documentation.

After upgrading to Tivoli Workload Scheduler version 8.4 and installing DB2 9.1, to migrate the version 8.3 data from DB2 8.2 or from Oracle, you should use one of the procedures documented in Administration and Troubleshooting.

The next sections describe the RDBMS-related prerequisites for the supported versions of DB2 and Oracle:

**DB2 Enterprise Server Edition**

DB2 Enterprise Server Edition is one of the two RDBMS options required for the installation of a master domain manager. Supported versions include:

- 8.2 (8.1.7) or later. For enhanced performance, use DB2 8.2.4 (8.1.11).
- 9.1

For a detailed list of databases for each supported operating system see Chapter 2, “Supported operating systems and requirements,” on page 27.

You do not need to install the DB2 Server on the same computer as the master domain manager. As Figure 5 on page 14 illustrates, when the master domain manager is installed on a computer other than the DB2 Enterprise Server Edition computer, DB2 administration client is needed to access the computer where the server is installed.
The database can be installed automatically when installing the master domain manager, or it can be preinstalled.

For information about the installation and configuration of DB2 Enterprise Server Edition and Client, refer to the relevant DB2 documentation on the DB2 CD.

If you want to install a master domain manager with DB2 client or server, make sure the DB2 Enterprise Server Edition prerequisites are met. For the DB2 supported platforms and patches, see platform\doc\language\install.htm on the DB2 CD.

When you are installing a Tivoli Workload Scheduler with an existing DB2 installation, the following must already be created:

- DB2 server host name
- DB2 server instance
- DB2 server port number
- DB2 server administrator user and password
- DB2 client administrator user and password
- Optionally, Tivoli Workload Scheduler DB2 client user and password that is used to access the DB2 server. The maximum length is 8 characters.

Make sure that your operating system meets the DB2 kernel requirements. If it does not, installation fails. You can use DB2 utilities for this, such as db2osconf.
When you are installing a master domain manager or a connector, the fully qualified domain name must be defined for the workstation otherwise the embedded version of WebSphere Application Server does not start and installation fails.

**Oracle Database**

Oracle Database is the other choice of RDBMS required to install a master domain manager. The Oracle database versions both for server and client required by Tivoli Workload Scheduler can be one of the following:

- Oracle Database 9i Release 2 - Enterprise Edition (9.2.0.x) or later
- Oracle Database 10g Release 2 - Enterprise Edition (10.2.0.x) or later

The Oracle Database must have the partitioning option installed.

The database must be already installed and configured before you install Tivoli Workload Scheduler. Before you start installation of a master domain manager, make sure you have the following information:

- The path pointing to where the Oracle installation is located. The path must identify a tree in the Oracle structure that includes the sqlplus executable.
- The net service name of the database. It must have been already created by the Oracle administrator.
- The name and password of the Oracle administrator.
- The name of the Tivoli Workload Scheduler data and temporary tablespaces. They must have been already created by the Oracle administrator.

**Installing DB2 on Solaris, AIX, and HP-UX**

When you are installing with a DB2 server or client on Solaris or AIX®, or a DB2 server on HP-UX you need to perform some preinstallation steps.

To set up DB2 installation for Solaris, AIX, or HP, perform the following steps:

1. Change to a temporary directory in a file system containing at least 1.5 gigabytes of free space by entering: cd /TMP, where /TMP represents the temporary directory.
2. If the product has the *.tar.Z extension, enter the following command to expand the installation image:

   ```
   zcat cdrom mount point/PRODUCT.tar.Z | tar -xvf -
   ```

   where PRODUCT is the DB2 product you are installing, and cdrom mount point is the directory where you have mounted the DB2 CD.
3. If the product has the *.tar.gz extension, enter the following command to expand the installation image:

   ```
   gunzip -c cdrom mount point/PRODUCT.tar.gz | tar -xvf -
   ```

   where PRODUCT is the DB2 product you are installing, and cdrom mount point is the directory where you have mounted the DB2 CD.

**Notes:**

1. `gunzip` is part of the AIX 5L™ default installation setup. If you do not have `gunzip`, install the `rpm.rte` file set from the AIX 5L installation media. The `rpm.rte` file set contains `gunzip`.
Planning installations

3. In addition to the software disk requirements, you need a file system with 1.5 gigabytes of free space to contain the tar.Z or tar.gz file and the expanded installation image.

Checking temporary directory requirements for installing using the installation wizard

Before starting an installation using the installation wizard, make sure that you meet the temporary directory space requirements. See Table 3.

Table 3. Temporary directories required size

<table>
<thead>
<tr>
<th>Component</th>
<th>Platform</th>
<th>Required space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engine</td>
<td>Windows</td>
<td>70 MB</td>
</tr>
<tr>
<td>Engine</td>
<td>UNIX and Linux</td>
<td>170 MB</td>
</tr>
<tr>
<td>Command-line client</td>
<td>Windows</td>
<td>20 MB</td>
</tr>
<tr>
<td>Command-line client</td>
<td>UNIX and Linux</td>
<td>40 MB</td>
</tr>
<tr>
<td>Agent</td>
<td>Windows</td>
<td>20 MB</td>
</tr>
<tr>
<td>Agent</td>
<td>UNIX and Linux</td>
<td>40 MB</td>
</tr>
</tbody>
</table>

The default temporary directories are /tmp for UNIX and Linux and the user temporary directory for Windows. For UNIX systems, If you decide to copy the installation images locally using setup.sh, you will need to consider additional disk space in /tmp.

Checking user authorization requirements

Depending on the installation method you choose, you need to check the authorization roles before beginning the install procedure.

Authorization roles for running the install, uninstall, and upgrade wizards

Table 4 provides the authorization roles required to use the installation wizard method of installation.

Table 4. Required authorization roles for running the installation wizard

<table>
<thead>
<tr>
<th>Activity</th>
<th>Required role</th>
</tr>
</thead>
</table>
| Installation using the wizard      | • Windows: your login account must be a member of the Windows Administrators group or domain administrators with Act as Part of the Operating System.  
• UNIX and Linux: root access      |
| Upgrade with an automatic data import | • Windows: your login account must be a member of the Windows Administrators group or domain administrators with Act as Part of the Operating System.  
• UNIX and Linux: root access  
• Write rights to the TWSUser/bin  
• Access, replace, and modify rights on all database objects that are to be imported into the new database |
Authorization roles for running data import after upgrade

Table 5 provides the authorization roles required to import the database objects after upgrade.

**Table 5. Required authorization roles for running the installation wizard**

<table>
<thead>
<tr>
<th>Activity</th>
<th>Required role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data import using the datamigrate command</td>
<td>• <strong>Windows</strong>: your login account must be a member of the Windows Administrators group or domain administrators with <em>Act as Part of the Operating System</em>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>UNIX and Linux</strong>: root access</td>
</tr>
<tr>
<td></td>
<td>• Write rights to the $TWSUser/bin$</td>
</tr>
<tr>
<td></td>
<td>• Access, replace, and modify rights on all database objects that are to be imported into the new database</td>
</tr>
</tbody>
</table>

Authorization roles for running the twsinst script

Table 6 provides the authorization roles required to use the twsinst method of installation.

**Table 6. Required authorization roles for running twsinst**

<table>
<thead>
<tr>
<th>Activity</th>
<th>Required role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Running the twsinst script</td>
<td>root access</td>
</tr>
</tbody>
</table>

Authorization roles for Software Distribution

Table 7 provides the authorization roles required to use the Software Distribution method of installation.

**Table 7. Required authorization roles for Software Distribution**

<table>
<thead>
<tr>
<th>Activity</th>
<th>Required role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Software Distribution to install a software package block</td>
<td>admin, senior, or super</td>
</tr>
<tr>
<td></td>
<td>• <strong>Windows</strong>: your login account must be a member of the Windows Administrators group with <em>Act as Part of the Operating System</em>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>UNIX and Linux</strong>: root access</td>
</tr>
</tbody>
</table>

**Tivoli Workload Scheduler user**

On UNIX and Linux operating systems, regardless of the method of installation you choose, the Tivoli Workload Scheduler user must be created manually before running the installation. Use the appropriate UNIX and Linux operating system commands to create the user.

**Note:** Some operating systems require that for users with a password, the password must be changed at the first login. When this is the case, do so otherwise the installation fails.

**Windows users domain rights and structure**

If you install on Windows operating systems, take into account the constraints listed next.
Tivoli Workload Scheduler user

Do not install two agents using the same user, one on local system and one defined in Microsoft Active Directory.

When you are installing:

• For a stand-alone workstation, the Windows user:
  – Is a member of the administrative group
  – Has the Act as part of the operating system privilege
  – Has the Log on as a service privilege, if the server is run as a service
  – Has the Impersonate a client after authentication right

• For a workstation that is a member of a domain, only a domain user can start the server process and:
  – Is a member of the domain administrative groups in the domain controller
  – Has the Act as part of the operating system privilege in the Domain Security Policy on the domain controller
  – Has the Act as part of the operating system privilege in the Local Security Policy on the local machine
  – Has the Log on as a service privilege on the local machine, if the server is run as a service
  – The user is a domain user and not a local user, which implies that when a machine is part of a domain, only a domain user can start the server

• For a domain controller workstation, the user:
  – Is a member of the domain administrative groups in the domain controller
  – Has the Act as part of the operating system privilege in the Domain Security Policy on the domain controller
  – Has the Act as part of the operating system privilege in the Domain Security Policy on the domain controller
  – Has the Log on as a service privilege on the domain controller, if the server is run as a service

Windows domain users therefore need the following permissions:

• Act as part of the operating system
• Log on:
  – Locally
  – As batch job
  – As service
• Replace process level token
• Impersonate a client after authentication right

Considerations for Windows domain controllers running Microsoft Active Directory

If you want to install Tivoli Workload Scheduler fault-tolerant agents on workstations where users that run jobs are domain users and the domain controller is running Microsoft Active Directory, you need to decide how to install the agents and configure the domain so that the jobmon process can obtain the correct information to let the users run jobs.

Before running a job, jobmon must retrieve information about the user running the job. If the user is a domain user and the domain controller is running Microsoft Active Directory, whether the user information can be retrieved depends on the information in the access control list (ACL) of that user. The main jobmon process that runs the job is started as the local system account (AUTHORITY\SYSTEM),
but it immediately impersonates the TWSuser that owns the fault-tolerant agent. This means that for jobmon to successfully launch the job, the TWSuser must have an access control entry (ACE) in the ACL of the user for which it is trying to retrieve information.

To resolve this issue, perform one of the following actions:

**Enable the TWSuser to access a set of users that run jobs.**
On the domain server, edit the ACL of all users that run jobs on the computer and add an ACE for the TWSuser for each. In this case, only the specified users can run the jobs submitted by jobmon.

**Allow all users to run jobs submitted by jobmon by using the TWS_BYPASS_DC system variable**
Create the TWS_BYPASS_DC system variable, with a value not null, and reboot the computer. In this case, jobmon obtains the user information without performing the security check for the ACE in the ACL of the user. All the local and the domain users can run the jobs submitted by jobmon.

**Allow all users to run jobs submitted by jobmon setting the TWSuser as a domain user**
Set up the TWSuser as a Windows domain user and install the instance of Tivoli Workload Scheduler using the TWSuser. In this case, all authenticated users on the domain controller can access the default ACL for a domain user. Jobs can then be launched by the local or the domain users. All the local and the domain users can run the jobs submitted by jobmon.

**Exclude the workstation from the security check on users ACL**
On the domain server, add the host name of the computer where the fault-tolerant agent is installed to the Pre-Windows 2000-Compatible Access Group. In this way from a security point of view, the domain controller interacts with this computer as if it was in a Windows domain which does not support Active Directory. In this case, all the local and the domain users can run the jobs submitted by jobmon. In addition, the domain controller does not prevent any local or domain user from running other processes not controlled by Tivoli Workload Scheduler.

### Checking environment settings for Windows Vista users

Before you install Tivoli Workload Scheduler on a Windows Vista computer that does not belong to a Windows domain, make sure that the computer name and the domain name are both registered in uppercase in the Windows environment settings. When the workstation is not in a Windows domain, the COMPUTERNAME and USERDOMAIN values are identical, but on Vista the USERDOMAIN value sometimes happens to be in lowercase even if the COMPUTERNAME is in uppercase.

To resolve this issue, perform the following actions:

1. Open a DOS command prompt shell
2. Run the `set` command to display the Windows environment settings
3. Check that the USERDOMAIN value is in uppercase. If this is not the case, follow this workaround to correct it:
4. Run the `set` command to change the value of COMPUTERNAME to a temporary host name of your choice:
   ```
   set /p COMPUTERNAME=MYTEMPHOST
   ```
5. Restart the system
6. Run the `set` command again as in step 4 replacing the temporary host name with the original one.

7. Restart the system

8. Check that the USERDOMAIN value is now in uppercase.

**Product instances**

Multiple copies of the product can be installed on a single computer provided that a unique name and installation path is used for each instance. Instances are recorded in the registry file. Former versions of Tivoli Workload Scheduler were also registered in the components file.

**Installing for end-to-end scheduling**

If you are installing Tivoli Workload Scheduler on a workstation used as a distributed agent for end-to-end scheduling, specify OPCMASTER as the name of the master domain manager during the installation process. For further information about installing for end-to-end scheduling, refer to *Tivoli Workload Scheduler Scheduling End-to-end*.

**Installation element criteria**

[Table 8] lists the criteria you must follow when installing. If you are installing using the installation wizard, most of these are checked by the system. However, when you are installing with another method they are not checked and installation fails if they are incorrect. Read these criteria before you begin installation.

<table>
<thead>
<tr>
<th>Element</th>
<th>Spaces</th>
<th>Max. Length</th>
<th>Valid Characters</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation Path</td>
<td>YES</td>
<td>–</td>
<td>Must not be numeric. For Windows, must be longer than 3 characters, the second character must be :, and the third character must be .. For UNIX and Linux operating systems must be longer than 1 character and the first character must be /. Tivoli Workload Scheduler for Applications version 8.2.1 or earlier cannot be installed on Tivoli Workload Scheduler version 8.4 if the directory path contains spaces. Parentheses () are not allowed.</td>
<td>C:\Program Files\IBM\TWS$(tws_user) UNIX and Linux User’s Home Directory</td>
</tr>
<tr>
<td>This Workstation Name</td>
<td>NO</td>
<td>16</td>
<td>The first character must be a letter.</td>
<td>HOST NAME</td>
</tr>
<tr>
<td>Master Domain Manager Name</td>
<td>NO</td>
<td>16</td>
<td>The first character must not be numeric.</td>
<td>MASTER</td>
</tr>
<tr>
<td>TCP/IP Port Number</td>
<td>NO</td>
<td>5</td>
<td>0 – 65535</td>
<td></td>
</tr>
<tr>
<td>Company Name</td>
<td>YES</td>
<td>40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS User Name</td>
<td>NO</td>
<td>16</td>
<td><code>abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789_</code> First character must be alphabetic</td>
<td></td>
</tr>
</tbody>
</table>
Table 8. Installation element validation criteria (continued)

<table>
<thead>
<tr>
<th>Element</th>
<th>Spaces</th>
<th>Max. Length</th>
<th>Valid Characters</th>
<th>Default</th>
</tr>
</thead>
</table>
| TWS User      | NO     | 31          | For FTAs on Windows: abcdefghijklmnopqrstuvwxyz
                                                                     ABCDEFGHIJKLMNOPQRSTUVWXYZ
                                                                     0123456789!?!_=^*_#]["$`-+;:,@  
                                                                     For Master, Backup Master, FTAs other platforms:  
                                                                     abcdefghijklmnopqrstuvwxyz
                                                                     ABCDEFGHIJKLMNOPQRSTUVWXYZ
                                                                     0123456789()=?+,:@                                    |
| Password      |        |             |                                                                                  |         |

For installation with DB2, make sure that the DB2 naming rules are met. For information about the DB2 naming rules, refer to the DB2 documentation.

The installation CDs

In each of the installation CDs, named as the supported platforms, you will find the following directories:

A platform specific installation image directory

The directory contains an acronym that identifies the platform. Values are:

- TWS_AIX
- TWS_HPUX
- TWS_HPIA64
- TWS_SOLARIS
- TWS_SOL_I386
- TWS_WIN
- TWS_WIN_X86_64
- TWS_LNX_I386
- TWS_LNX_PPC
- TWS_LNX_S390
- TWS_LNX_X86_64
- TWS_OTHER

RESPONSEFILES

containing all the response files for a silent installation

launchpad

containing launchpad specific code

API

containing the Javadoc and client jar files

license

containing the license agreements

TWS4ITM

containing files for the integration with IBM Tivoli Monitoring (except for HPUX/IA64 Itanium®).

TWSPUGINS

containing the Tivoli Workload Scheduler plug-ins (except for HPUX/IA64 Itanium).

The following Tivoli Workload Scheduler Version 8.4 CDs are required to start the installation process:

Tivoli Workload Scheduler Engine for AIX

Includes images for the AIX operating system
Installation validation criteria

- **Tivoli Workload Scheduler Engine for HP-UX on PA-RISC**
  Includes images for the HPUX on PA-RISC operating system.

- **Tivoli Workload Scheduler Engine for HP-UX on Itanium**
  Includes images for the HPUX/IA64 (Itanium) operating system.

- **Tivoli Workload Scheduler Engine for Solaris SPARC**
  Includes images for the Solaris SPARC operating system.

- **Tivoli Workload Scheduler Engine for Solaris x64**
  Includes images for the Solaris x64 operating system.

- **Tivoli Workload Scheduler Engine for Windows**
  Includes images for the Windows operating system.

- **Tivoli Workload Scheduler Engine for Windows x64**
  Includes images for the Windows x64 operating system.

- **Tivoli Workload Scheduler Engine for Linux on POWER™**
  Includes images for the Linux PPC operating system.

- **Tivoli Workload Scheduler Engine for Linux on x86**
  Includes images for the Linux x86 operating system.

- **Tivoli Workload Scheduler Engine for Linux on System z9 and zSeries®**
  Includes images for Linux z9™ and zSeries operating system.

- **Tivoli Workload Scheduler Engine for Linux on x86-64**
  Includes images for the Linux x86_64 operating system.

- **Tivoli Workload Scheduler Integrations and Engine for i5/OS®**
  Includes the following directories:
  - i5_05 containing images for the i5/OS operating system.
  - TDW_Enablement_pack containing the files needed for the integration with the Tivoli Data Warehouse product
  - TWS4ITM containing the files for the integration with IBM Tivoli Monitoring 6.1
  - TEPClientConfig containing the files needed for the integration with Tivoli Enterprise™ Portal
  - TWSPLUGINS containing the Tivoli Workload Scheduler plug-ins
  - 1icense containing license agreements

**Note:** When you copy the image of a specific operating system onto the workstation for installation using the wizard, you must copy the complete CD contents to the drive from where you run installation. When the drive is a UNC mapped drive, the remote path must be mapped to a drive on the installation computer. For a complete list of the supported operating systems and their prerequisites, see “Supported operating systems” on page 27.

**Installation paths**

The installation installs Tivoli Workload Scheduler files for the *TWSuser* in *TWShome*, where:

- **TWShome**
  The installation location. On Windows operating systems, the default installation location is defined as `C:\Program Files\IBM\TWS\TWSuser`, but you can specify a different location. On UNIX and Linux operating systems, the product is installed in the user’s home directory.
**Installation log files**

Details of the installation process are logged in log files located in the temporary directory set on the local computer. You can check the following log files for information about the installation.

- **twsismp.log**  The trace file to which the installation wizard writes.
- **summary.log**  The log file to which the installation wizard writes.
- **FP_TWS_{(operating_system)}_{(user)}^8.4.00.log**  The file to which Software Distribution writes.
- **TWS_{(operating_system)}_{(TWSuser)}^8.4.00actiontools.log**  The file to which Software Distribution writes pre- and post-installation actions.
- **twsinst_{operating_system}_sm3_50^8.4.00.log**  The file to which twsinst writes.
- **datamigrate_{object}_nnnnnn.log**  The file to which data import writes, where object is the object being imported and nnnnn is the run number of the data import.

For more information about log files, refer to *Tivoli Workload Scheduler Administration and Troubleshooting*.

**Windows services registration**

An installation on Windows operating systems registers the following services with the Windows Service Control Manager:

- Tivoli Workload Scheduler (for TWSuser)
- Tivoli Netman (for TWSuser)
- Tivoli Token Service (for TWSuser)
- Autotrace Runtime
- Websphere Application Server (for TWSuser)

The Service Control Manager maintains its own user password database. Therefore, if the TWSuser password is changed following installation, you must use the Services applet in the Control Panel to assign the new password for the Tivoli Token Service and Tivoli Workload Scheduler (for TWSuser).

**Modifying jobmon service rights for Windows**

On Windows systems, the Tivoli Workload Scheduler jobmon service runs in the SYSTEM account with the right Allow Service to Interact with Desktop granted to it. You can remove the right for security reasons. However, this prevents the service from launching interactive jobs that run in a window on the user’s desktop. These jobs are not accessible and do not have access to desktop resources. As a result, they may run forever or abend due to lack of resources.

**Symlink options**

Several of the installation methods discuss the symlink options. The link option determines the path used to create links to Tivoli Workload Scheduler’s utility commands. Table 9 lists the available options.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>twshome/bin/at</td>
<td>usr/bin/mat</td>
</tr>
<tr>
<td>twshome/bin/batch</td>
<td>usr/bin/mbatch</td>
</tr>
</tbody>
</table>
Table 9. Symlink options (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>twshome/bin/dtcalc</td>
<td>/usr/bin/dtcalc</td>
</tr>
<tr>
<td>twshome/bin/jobstdl</td>
<td>/usr/bin/jobstdl</td>
</tr>
<tr>
<td>twshome/bin/maestro</td>
<td>/usr/bin/maestro</td>
</tr>
<tr>
<td>twshome/bin/mdemon</td>
<td>/usr/bin/mdemon</td>
</tr>
<tr>
<td>twshome/bin/morestdl</td>
<td>/usr/bin/morestdl</td>
</tr>
<tr>
<td>twshome/bin/muser</td>
<td>/usr/bin/muser</td>
</tr>
<tr>
<td>twshome/bin/parms</td>
<td>/usr/bin/parms</td>
</tr>
</tbody>
</table>

### Engine coexistence and upgrade notes

This section contains information about coexistence with older versions and upgrade possibilities.

#### Coexistence with previous versions

The Tivoli Workload Scheduler version 8.4 distributed engine can be installed on any computer containing a prior version of Tivoli Workload Scheduler, provided that both the TWSuser and the installation path are different from those of the previous versions.

#### Upgrading existing versions

The upgrade of a Tivoli Workload Scheduler network can be performed top-down or bottom-up. The advantages and disadvantages of these approaches are discussed in the upgrade chapters in this document.

Table 10 shows the versions of Tivoli Workload Scheduler components that can be upgraded to version 8.4:

Table 10. Upgrade availability for Tivoli Workload Scheduler components

<table>
<thead>
<tr>
<th>Component</th>
<th>Version</th>
<th>Recommended fix pack level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tivoli Workload Scheduler engine (master domain manager, backup master domain manager, fault-tolerant agent)</td>
<td>8.2</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>8.2.1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>8.3</td>
<td>3</td>
</tr>
<tr>
<td>Connector</td>
<td>Upgrade available from version 8.3</td>
<td></td>
</tr>
<tr>
<td>Job Scheduling Console</td>
<td>Upgrade available from version 8.3</td>
<td></td>
</tr>
</tbody>
</table>

- If you are using Tivoli Workload Scheduler components at a lower fix pack level, install the fix pack at the indicated level, or higher.
- If you are using Tivoli Workload Scheduler components from a previous version not supported in Table 10, consider these options:
  - Replace them with a fresh installation of the version 8.4 component
  - Upgrade them as follows:
    1. Upgrade them first to one of the supported upgrade platforms, using the upgrade programs and procedures documented for that platform
    2. Apply the necessary fix packs, as shown in Table 10
3. Upgrade them to version 8.4.
Symlink options
Chapter 2. Supported operating systems and requirements

This chapter contains information about the software and hardware prerequisites for Tivoli Workload Scheduler. It is divided into the following sections:

- “Supported operating systems”
- “Software requirements” on page 30
- “Hardware requirements” on page 36

Supported operating systems

This chapter lists the platforms that are supported by the engine.

Table 11 identifies the operating systems supported (where the support is not full, details are supplied in the notes).

The following acronyms are used:

**MDM** Master domain manager

**BKM** Backup master domain manager

**FTA** Fault-tolerant agent (since version 8.3, the concept of separate fault-tolerant and standard agents has been replaced by a single agent component, which is fault-tolerant)

**CONN** Connector for a distributed engine

**CLI client** Command-line client

**JSC** Job Scheduling Console

<table>
<thead>
<tr>
<th>Platforms</th>
<th>MDM/ BKM</th>
<th>FTA</th>
<th>CONN</th>
<th>CLI client</th>
<th>JSC (see notes 2 and 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM AIX 5L versions 5.2, 5.3</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>HP-UX version 11i v1 PA-RISC</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>HP-UX version 11i v2 Itanium and PA-RISC</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>HP-UX version 11i v3 PA-RISC</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>HP-UX version 11i v3 Itanium</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Solaris Operating Environment version 9</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Solaris Operating Environment version 10</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Solaris Operating Environment version 10 AMD (Opteron)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Microsoft Windows Server 2003: Standard, Enterprise, and Data Center</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Microsoft Windows Server 2003: Standard, Enterprise, Data Center for 64-bit Itanium2 based systems - Supported in tolerance mode only (32-bit)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Microsoft Windows Server 2003: Standard and Enterprise for AMD64 and EMT64T Kernel 64 Supported in tolerance mode only (32 bit)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
## Engine operating systems

### Table 11. Engine supported platforms (continued)

<table>
<thead>
<tr>
<th>Platforms</th>
<th>MDM/BKM</th>
<th>FTA</th>
<th>CONN</th>
<th>CLI client</th>
<th>JSC (see notes 2 and 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows XP Professional with SP2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Microsoft Windows Vista</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Microsoft Windows Vista for AMD64 and EMT64T</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM i5/OS versions 5.3, 5.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 3.0 (AS/ES): System x® (IA32) and eSeries (AMD64 and EM64T) Kernel 32 (see note 4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 3.0 (AS/ES): System i™ Kernel 64 (see note 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 3.0 (AS/ES): System p® Kernel 64 (see note 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 3.0 (AS/ES): System z Kernel 32 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 4.0 (AS/ES): System z Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 4.0 (AS/ES): System x and eSeries (AMD64 and EM64T) Kernel 32 (see note 4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 4.0 (AS/ES): System x (AMD64 and EM64T) Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 4.0 (AS/ES): System p Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 4.0 (AS/ES): System z Kernel 32 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System x and eSeries (AMD64 and EM64T) Kernel 32 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System x (AMD64 and EM64T) Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System i Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System p Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System z Kernel 32 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Red Hat Enterprise Linux 5.0: System z Kernel 64 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SuSe Linux Enterprise Server 9 and 10: System x (IA32) and eSeries (AMD64 and EM64T) Kernel 32 (see notes 1 and 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SuSe Linux Enterprise Server 9 and 10: System i, Kernel 64 (see notes 1 and 3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SuSe Linux Enterprise Server 9 and 10: System z, Kernel 32 (see note 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 11. Engine supported platforms (continued)

<table>
<thead>
<tr>
<th>Platforms</th>
<th>MDM/ BKM</th>
<th>FTA</th>
<th>CONN</th>
<th>CLI client</th>
<th>JSC (see notes 2 and 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuSe Linux Enterprise Server 9 and 10: System z, Kernel 64 (see note 4)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>SuSe Linux Enterprise Server 9 and 10: System p, Kernel 64 (see notes 1 and 3)</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>

Notes:

1. InstallShield installation on Linux operating systems needs the `bc` utility to be installed. For further information see http://community.installshield.com/archive.

2. The Job Scheduling Console is included in this table to aid you for planning purposes. Its hardware and software requirements are documented in the Tivoli Workload Scheduler: Job Scheduling Console User's Guide.

3. The Job Scheduling Console is not supported on SuSe Linux Enterprise Server 10.

4. On Red Hat Linux 3.0 DB2 9.1 is not supported. On these systems, you need to install DB2 8.2 manually before running the Tivoli Workload Scheduler installation.

**Supported operating system matrix**

IBM maintains a matrix showing the supported operating systems for all its Tivoli products: [http://www-1.ibm.com/support/docview.wss?rs=203&uid=swg21067036](http://www-1.ibm.com/support/docview.wss?rs=203&uid=swg21067036)

Use the link to check the latest supported operating system information.

**Supported platforms for the prerequisite database**

Tivoli Workload Scheduler requires the use of either the DB2 or the Oracle relational database. The supported platforms are as follows:

**Supported operating systems for the prerequisite version of DB2**

Tivoli Workload Scheduler requires the minimum versions of DB2 indicated in Table 12.

Table 12. Minimum required versions of DB2

<table>
<thead>
<tr>
<th>Platform</th>
<th>DB2 minimum version</th>
</tr>
</thead>
<tbody>
<tr>
<td>All operating systems</td>
<td>8.2 (8.1 with fix pack 7) or later</td>
</tr>
</tbody>
</table>

**Bundled DB2:** Tivoli Workload Scheduler is supplied with a version of the DB2 server and client for the supported platforms. This version can either be installed by you separately before the product installation, or automatically as part of the product installation. Refer to Table 14 on page 33 for details on the DB2 bundled versions for each of the supported operating systems.

**Note:** The bundled version of DB2 is provided solely for use with Tivoli Workload Scheduler (see the product license for details).
## Supported platforms for the database

### Supported operating systems for the prerequisite version of Oracle

Tivoli Workload Scheduler requires the use of the following versions of Oracle with the partitioning option installed:

- Oracle Database 9i Release 2 - Enterprise Edition (9.2.0.x) or later
- Oracle Database 10g Release 2 - Enterprise Edition (10.2.0.x) or later.

The Oracle database software is not supplied with Tivoli Workload Scheduler. You must use your own version of the Oracle database software.

For a complete list of supported operating systems for Oracle see the Oracle documentation.

### Supported operating systems for the bundled application server

Tivoli Workload Scheduler uses the embedded version of Websphere Application Server, version 6.1. All platforms supported by Tivoli Workload Scheduler are supported by the application server. It is installed automatically. In the information relating to hardware and memory requirements in this document, the application server is treated as part of Tivoli Workload Scheduler, and its requirements are not listed separately.

**Note:** The embedded version of Websphere Application Server is provided solely for use with Tivoli Workload Scheduler (see the product license for details). This version has no limit on the number of cpus on which it can be installed.

## Software requirements

### Software requirements for Tivoli Workload Scheduler

Table 13 lists the software requirements for each supported operating system.

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Minimum required patch level</th>
<th>Additional prerequisites</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM AIX</td>
<td>AIX 5L 5.2 with Recommended Maintenance package 5200-07</td>
<td>For HP-UX 11i v1 (11.11) the following patches:</td>
</tr>
<tr>
<td></td>
<td>AIX 5L 5.3 with Service Pack 5300-04-01</td>
<td>PHNE_35351,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PHNE_35765,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IPv6NCF11i_B.11.11.0705_HP_UX_B.11.11_32+64.depot.</td>
</tr>
<tr>
<td>Operating System</td>
<td>Minimum required patch level</td>
<td>Additional prerequisites</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>HP-UX Itanium (HPIA64)</td>
<td>HP-UX 11iiv1 (11.11) with Standard Patch Bundles of March 2006&lt;br&gt;HP-UX 11iiv2 (11.23) with Standard Patch Bundles of March 2006 and Sep 2004 Bundle 11i + Java patches PHCO_30476 and PHSS_34201 (64 bit kernel support)&lt;br&gt;HP-UX 11iiv3 (11.31)</td>
<td>For HP-UX 11i v1 (11.11)&lt;br&gt;the following patches: PHNE_35351, PHNE_35765,&lt;br&gt;IPv6NCF11i_B.11.11.0705_HP_UX_&lt;br&gt;B.11.11..32 +64.depot.</td>
</tr>
<tr>
<td>Linux IA32</td>
<td>Red Hat Enterprise Linux AS, Version 3 with Update 5 or 6&lt;br&gt;Red Hat Enterprise Linux AS, Version 4 with Update 2&lt;br&gt;Red Hat Enterprise Linux ES, Version 3 with Update 5 or 6&lt;br&gt;Red Hat Enterprise Linux ES, Version 4 with Update 2&lt;br&gt;Red Hat Enterprise Linux, Version 5&lt;br&gt;SUSE Linux Enterprise Server, Version 9 with SP2 or 3&lt;br&gt;SUSE Linux Enterprise Server, Version 10</td>
<td>For all Linux IA32 operating systems:&lt;br&gt;libstdc++.so.5 and libgcc_s.so.1 are required.&lt;br&gt;libstdc+-libc6.1-2.so.3 is required, alternatively a soft link to libstdc+-libc6.2-2.so.3 is needed.&lt;br&gt;For Red Hat Enterprise Linux, version 5: libXp.so.6 and libXmu.so.6 )</td>
</tr>
<tr>
<td>Linux X86_64</td>
<td>Red Hat Enterprise Linux AS, Version 3 with Update 5 or 6&lt;br&gt;Red Hat Enterprise Linux AS, Version 4 with Update 2&lt;br&gt;Red Hat Enterprise Linux ES, Version 3 with Update 5 or 6&lt;br&gt;Red Hat Enterprise Linux ES, Version 4 with Update 2&lt;br&gt;Red Hat Enterprise Linux, Version 5&lt;br&gt;SUSE Linux Enterprise Server, Version 9 with SP2 or 3&lt;br&gt;SUSE Linux Enterprise Server, Version 10</td>
<td>For all Linux X86_64 operating systems:&lt;br&gt;libstdc++.so.5 and libgcc_s.so.1 are required.&lt;br&gt;libstdc+-libc6.1-2.so.3 is required, alternatively a soft link to libstdc+-libc6.2-2.so.3 is needed.&lt;br&gt;(linux 32 and 64 bit kernel support)&lt;br&gt;For Red Hat Enterprise Linux, version 5: libXp.so.6 and libXmu.so.6 (32 and 64 bit kernel support)</td>
</tr>
<tr>
<td>Linux System z</td>
<td>Red Hat Enterprise Linux AS, Version 3 with Update 5 or 6 (31 bit kernel only)&lt;br&gt;Red Hat Enterprise Linux AS, Version 4 with Update 2&lt;br&gt;Red Hat Enterprise Linux, Version 5&lt;br&gt;SUSE Linux Enterprise Server, Version 9 with SP2 or 3&lt;br&gt;SUSE Linux Enterprise Server, Version 10</td>
<td>For all System z platforms:&lt;br&gt;libstdc++.so.5 and libgcc_s.so.1 are required.&lt;br&gt;libstdc+-libc6.1-2.so.3 is required, alternatively a soft link to libstdc+-libc6.2-2.so.3 is needed.&lt;br&gt;(linux 32 and 64 bit kernel support)&lt;br&gt;For Red Hat Enterprise Linux, version 5: libXp.so.6 and libXmu.so.6</td>
</tr>
</tbody>
</table>
## Software requirements

**Table 13. Software Requirements for Tivoli Workload Scheduler (continued)**

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Minimum required patch level</th>
<th>Additional prerequisites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux PPC</td>
<td>Red Hat Enterprise Linux AS, Version 3.0 with Update 5 or 6 (32 bit kernel only)</td>
<td>For all Linux PPC operating systems: libstdc++.so.5 and libgcc_s.so.1 are required. libstdc++-libc6.1-2.so.3 is required, alternatively a soft link to libstdc++-libc6.2-2.so.3 is needed. (linux 32 and 64 bit kernel support)</td>
</tr>
<tr>
<td></td>
<td>Red Hat Enterprise Linux AS, Version 4 with Update 2</td>
<td>For Red Hat Enterprise Linux, version 5: libXp.so.6 and libXmu.so.6 (32 and 64 bit kernel support)</td>
</tr>
<tr>
<td></td>
<td>Red Hat Enterprise Linux, Version 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUSE Linux Enterprise Server, Version 9 with SP2 or 3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUSE Linux Enterprise Server, Version 10</td>
<td></td>
</tr>
<tr>
<td>Linux and i5/OS System i</td>
<td>i5/OS and OS/400®, Version V5R3</td>
<td>For all Linux System i platforms: libstdc++.so.5 and libgcc_s.so.1 are required. libstdc++-libc6.1-2.so.3 is required, alternatively a soft link to libstdc++-libc6.2-2.so.3 is needed. (linux 32 and 64 bit kernel support)</td>
</tr>
<tr>
<td></td>
<td>i5/OS, Version V5R4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Red Hat Enterprise Linux AS, Version 3.0 with Update 5 or 6 (32 bit kernel only)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Red Hat Enterprise Linux AS, Version 4 with Update 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Red Hat Enterprise Linux, Version 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUSE Linux Enterprise Server, Version 9 with SP2 or 3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SUSE Linux Enterprise Server, Version 10</td>
<td></td>
</tr>
<tr>
<td>Solaris SPARC</td>
<td>Sun Solaris operating environment, Version 9 with the latest Patch Cluster</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sun Solaris operating environment, Version 10 with the latest Patch Cluster</td>
<td></td>
</tr>
<tr>
<td>Solaris i386 (Opteron)</td>
<td>Sun Solaris operating environment, Version 10 with the latest Patch Cluster</td>
<td>Patch 119964-05</td>
</tr>
<tr>
<td>Windows</td>
<td>Microsoft Windows Vista Enterprise</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows Server 2003, Datacenter with SP1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows Server 2003, Enterprise with SP1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows Server 2003, Standard with SP1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows XP Professional with SP2</td>
<td></td>
</tr>
<tr>
<td>Windows x86_64</td>
<td>Microsoft Windows Server 2003 x64 Editions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Windows XP Professional with SP2 x64 Editions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Microsoft Vista Enterprise x64 Editions</td>
<td></td>
</tr>
</tbody>
</table>
### Software requirements for the bundled version of DB2

Table 14 lists the software requirements for the bundled version of DB2. For specific platform limitations see “Notes” on page 36.

**Table 14. Software requirements for the bundled DB2**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Bundled DB2 Version</th>
<th>Software Prerequisites</th>
</tr>
</thead>
</table>
| IBM AIX  | DB2 Version 9.1      | AIX 5.2  
64-bit AIX kernel is required  
Technology Level (TL) 5200-08 and  
Service Pack (SP) 5200-08-02  
Minimum C++ runtime level is  
xIC.rte 8.0.0.4 and xIC.aix50.rte 8.0.0.8  
AIX 5.3:  
64-bit AIX kernel is required  
Technology Level (TL) 5300-04 and  
Service Pack (SP) 5300-04-02  
Minimum C++ runtime level is  
xIC.rte 8.0.0.4 and xIC.aix50.rte 8.0.0.8  |
| HP-UX PA-RISC and Itanium | DB2 Version 9.1 Fix Pack 1 for PA-RISC platform.  
DB2 Version 9.1 Fix Pack 3 for ITANIUM platform | HP-UX 11i v3  
For Itanium based systems only  
HP-UX 11i v2 (11.23.0505) operating systems with:  
May 2005 Base Quality (QPKBASE) bundle May 2005 Applications Quality (QPAPPS) bundle and the PHNE_32606 patch. (64-bit HP-UX kernel is required; server only) |
| Linux x86 | DB2 Version 9.1 Fix Pack 3 | Red Hat Enterprise Linux 4 Update 3  
Base Kernel Level: 2.6.9 Libraries:  
glibc-2.3.4  
Red Hat Enterprise Linux 5 GA  
Base Kernel Level: 2.6.18 Libraries:  
lstdc++.so.5  
SUSE Linux Enterprise Server 9 with SP3  
Base Kernel Level: 2.6.5 Libraries:  
glibc-2.3.3  
Suse Linux Enterprise Server 10 GA  
Base Kernel Level: 2.6.16 Libraries:  
glibc-2.4.31 |
### Requirements for bundled DB2

**Table 14. Software requirements for the bundled DB2 (continued)**

<table>
<thead>
<tr>
<th>Platform</th>
<th>Bundled DB2 Version</th>
<th>Software Prerequisites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux X86_64</td>
<td>DB2 Version 9.1 Fix Pack 3</td>
<td><strong>Red Hat Enterprise Linux 4 Update 3</strong>&lt;br&gt;Base Kernel Level: 2.6.9 Libraries: glibc-2.3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Red Hat Enterprise Linux 5 Version 5</strong>&lt;br&gt;Base Kernel Level: 2.6.18 Libraries: libstdc++.so.5 (32 and 64 bit kernel support)</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SUSE Linux Enterprise Server 9 with SP3</strong>&lt;br&gt;Base Kernel Level: 2.6.5 Libraries: glibc-2.3.3 (32 and 64 bit kernel support)</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SUSE Linux Enterprise Server, version 10</strong>&lt;br&gt;Base Kernel Level: 2.6.16 Libraries: glibc-2.4.31 (32 and 64 bit kernel support)</td>
</tr>
<tr>
<td>Linux System z</td>
<td>DB2 Version 9.1 Fix Pack 3</td>
<td><strong>Red Hat Enterprise Linux 4 Update 3</strong>&lt;br&gt;Base Kernel Level: 2.6.9 Libraries: glibc-2.3.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Red Hat Enterprise Linux 5 Version 5</strong>&lt;br&gt;Base Kernel Level: 2.6.18 Libraries: libstdc++.so.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SUSE Linux Enterprise Server 9 with SP3</strong>&lt;br&gt;Base Kernel Level: 2.6.5 Libraries: glibc-2.3.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>SUSE Linux Enterprise Server, version 10</strong>&lt;br&gt;Base Kernel Level: 2.6.16 Libraries: glibc-2.4.31</td>
</tr>
</tbody>
</table>
Table 14. Software requirements for the bundled DB2 (continued)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Bundled DB2 Version</th>
<th>Software Prerequisites</th>
</tr>
</thead>
</table>
| Linux PPC          | DB2 Version 9.1 Fix Pack 3 | Red Hat Enterprise Linux 4 Update 3  
Base Kernel Level: 2.6.9 Libraries: glibc-2.3.4 Additional prerequisite: IBM XL C/C++ Advanced Edition V8.0 for Linux Runtime Environment Component |
|                    |                     | Red Hat Enterprise Linux 5 Version 5  
Base Kernel Level: 2.6.18 Libraries: libstdc++.so.5 Additional prerequisite: IBM XL C/C++ Advanced Edition V8.0 for Linux Runtime Environment Component |
|                    |                     | SUSE Linux Enterprise Server 9 with SP3  
POWER 5 hardware is supported.  
Base Kernel Level: 2.6.5 Libraries: glibc-2.3.3 Additional prerequisite: IBM XL C/C++ Advanced Edition V8.0 for Linux Runtime Environment Component |
|                    |                     | SUSE Linux Enterprise Server, version 10  
POWER 5 hardware is supported.  
Base Kernel Level: 2.6.16 Libraries: glibc-2.4.31 Additional prerequisite: IBM XL C/C++ Advanced Edition V8.0 for Linux Runtime Environment Component |
| Solaris SPARC      | DB2 Version 9.1     | Solaris Operating environment Version 10 |
|                    |                     | Solaris Operating environment Version 9  
64-bit kernel Patches 111711-12 and 111712-12  
Kernel parameters may need to be updated, run the db2osconf utility and refer to the DB2 documentation for details. |
| Solaris i386       | DB2 Version 9.1     | Solaris Operating Environment Version 10  
Kernel parameters may need to be updated, run the db2osconf utility and refer to the DB2 documentation for details. |
Requirements for bundled DB2

Table 14. Software requirements for the bundled DB2 (continued)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Bundled DB2 Version</th>
<th>Software Prerequisites</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Windows 2003 Enterprise Edition (32-bit and 64-bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Windows 2003 Datacenter Edition (32-bit and 64-bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Service Pack 1 or later R2 is also supported Windows 2003 Service Pack 2 is required on systems with dual-core</td>
</tr>
</tbody>
</table>

Notes:
1. On Red Hat Linux 3.0 DB2 9.1 is not supported. On these systems, you need to install DB2 8.2 manually before running the Tivoli Workload Scheduler installation.
2. DB2 9.1 Fix Pack 3 is supported on HP-UX 11i v3 for Itanium platforms only. For Pa-RISC, DB2 9.1 Fix Pack 1 is required.

Supported LDAP Servers

The following is a list of the LDAP servers supported by Tivoli Workload Scheduler:
- IBM Tivoli Directory Server 5.1, 5.2 or 6.0
- IBM z/OS Security Server 1.4, 1.5 or 1.6
- IBM z/OS.e Security Server 1.4, 1.5 or 1.6
- Windows Active Directory 2003
- Sun ONE DS

Hardware requirements

This section describes the hardware requirements for Tivoli Workload Scheduler and includes information such as machine types and disk space.

Tivoli Workload Scheduler consists of the following components and optional features:
- The engine, configured as a master (MDM), backup master (BKM), fault-tolerant agent (FTA)
- The command-line client (CLI)
- Connector for the distributed engine (Conn)
- Tivoli Workload Scheduler database (DB), on DB2 or Oracle

Note: Language packs and license agreement files are included.
Disk space requirements

Table 15 summarizes the minimum disk space requirements of the Tivoli Workload Scheduler components and features on supported operating systems.

Table 15. Disk space requirements, in megabytes, for Tivoli Workload Scheduler components and features

<table>
<thead>
<tr>
<th>Operating system</th>
<th>MDM with DB2 server (see note 1)</th>
<th>MDM/ BKM with DB2 client (see note 2)</th>
<th>MDM/ BKM (see note 2)</th>
<th>DB2 server stand-alone (see note 3)</th>
<th>FTA</th>
<th>CLI client</th>
<th>CONN</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX</td>
<td>1375</td>
<td>475</td>
<td>540</td>
<td>240</td>
<td>210</td>
<td>90</td>
<td>330</td>
</tr>
<tr>
<td>HP-UX</td>
<td>1595</td>
<td>655</td>
<td>555</td>
<td>240</td>
<td>275</td>
<td>90</td>
<td>280</td>
</tr>
<tr>
<td>Linux</td>
<td>1245</td>
<td>500</td>
<td>530</td>
<td>240</td>
<td>180</td>
<td>90</td>
<td>350</td>
</tr>
<tr>
<td>Solaris</td>
<td>1525</td>
<td>600</td>
<td>600</td>
<td>240</td>
<td>210</td>
<td>90</td>
<td>390</td>
</tr>
<tr>
<td>Solaris I386</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>Windows</td>
<td>855</td>
<td>465</td>
<td>440</td>
<td>240</td>
<td>110</td>
<td>90</td>
<td>330</td>
</tr>
<tr>
<td>i5/OS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>30</td>
</tr>
</tbody>
</table>

Notes:
1. These figures include the presence of the base structure for Tivoli Workload Scheduler in the DB2 database.
2. These figures do not include the presence of the base structure for Tivoli Workload Scheduler in the DB2 database, because this is indicated in either the entry for the MDM with a DB2 server, or the entry for the standalone DB2.
3. The space listed in this column is divided as follows:
   • 200 megabytes in the DB2 installation path
   • 20 megabytes in the Data tablespace path
   • 20 megabytes in the Reports tablespace path
4. This table does not include the space required to operate Tivoli Workload Scheduler. For this you must consider the additional space required for the database data, the Symphony file, the log files, temporary files that are placed on the local hard drive, and files associated with the jobs managed by Tivoli Workload Scheduler. You should also consider the archive period for the Symphony file and the retention period for log files.
5. Table 15 does not include the space requirements for the temporary installation files, details of which are given in Table 16.

Table 16. Temporary file space, in megabytes, required by the installation

<table>
<thead>
<tr>
<th>Operating system type</th>
<th>MDM/BKM</th>
<th>FTA</th>
<th>CLI client</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIX</td>
<td>170</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>Windows</td>
<td>70</td>
<td>20</td>
<td>20</td>
</tr>
</tbody>
</table>

6. Table 16 does not include the requirements for the virtual memory file (sometimes called page file or swap space). The installation requires at least 512 MB of virtual memory on any operating system.
7. For information about Oracle permanent and temporary file space requirements see the Oracle documentation.
Hardware requirements

Recommended memory requirements

Table 17 summarizes the memory requirements for the various components, on all platforms:

Table 17. Memory requirements, in megabytes

<table>
<thead>
<tr>
<th>Required / recommended</th>
<th>MDM/BKM</th>
<th>FTA</th>
<th>CLI client</th>
<th>CONN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommended</td>
<td>2048</td>
<td>256</td>
<td>256</td>
<td>1536</td>
</tr>
<tr>
<td>Required</td>
<td>1024</td>
<td>256</td>
<td>256</td>
<td>1024</td>
</tr>
</tbody>
</table>

For information about Oracle memory and swap file space requirements see the Oracle documentation.
Part 2. Installing, upgrading, and uninstalling
Chapter 3. Installing a fresh Tivoli Workload Scheduler

This chapter describes how to perform a fresh installation of Tivoli Workload Scheduler version 8.4. It is divided into the following sections:

- “Installing Tivoli Workload Scheduler using the installation wizard” on page 42
- “Installing Tivoli Workload Scheduler agents using twsinst” on page 61
- “Installing Tivoli Workload Scheduler agents using Software Distribution” on page 63

Selecting your installation method

There are several installation methods you can use to install Tivoli Workload Scheduler:

**InstallShield wizard**

Tivoli Workload Scheduler master domain managers, backup masters, agents, and connectors can be installed by launching the individual setup files for each supported platform.

You can use the installation wizard in interactive or silent mode. In interactive mode, the wizard guides you through the installation steps. In silent mode a response file provides the relevant information to the install process, which is run in background.

This method of installation uses a Java Virtual Machine, and therefore has specific system requirements. See Chapter 2, “Supported operating systems and requirements,” on page 27 for details on installation requirements.

**Launchpad**

You can use the launchpad to guide you through the installation of all the Tivoli Workload Scheduler components from a single interface.

From the launchpad you can:

- Install or upgrade all Tivoli Workload Scheduler components
- Install the Job Scheduling Console
- Install the Tivoli Dynamic Workload Console
- Access information on Tivoli Workload Scheduler

The launchpad automatically accesses and runs the related installation setup file in interactive mode.

**twsinst script for UNIX and Linux operating systems**

Tivoli Workload Scheduler agents can be installed on UNIX and Linux operating systems using the twsinst script. This method does not use a Java Virtual Machine and can be used instead of the installation wizard. See “Installing Tivoli Workload Scheduler agents using twsinst” on page 61.

**Software Distribution software package blocks (SPBs)**

Tivoli Workload Scheduler agents can be installed using the Software Distribution component of IBM Tivoli Configuration Manager, versions 4.1, 4.2, 4.2.1, 4.2.2, or 4.2.3 by distributing software package blocks. See “Installing Tivoli Workload Scheduler agents using Software Distribution” on page 63.

Note: Refer to Chapter 2, “Supported operating systems and requirements,” on page 27 for a list of supported operating systems.
Installing using the installation wizard

Installing Tivoli Workload Scheduler using the installation wizard

This section describes how to install Tivoli Workload Scheduler components using the installation wizard. The installation wizard runs on all supported operating systems. Refer to "Supported operating systems" on page 27 for a complete list of supported operating systems.

Before you begin

Before you begin your installation using the installation wizard, consider the following checklist. Some items might apply to your specific environment. This list provides a quick solution to simple environment related problems.

- If you are installing on HP-UX operating systems:
  - Make sure the MAXDSIZ parameter is set to a minimum of 128 MB.
- If you are using Windows Terminal Services:
  - Set the install user with the command: change user /install.
- On UNIX and Linux operating systems, if you get an error message indicating insufficient space for the installation wizard temporary data in the default /tmp directory:
  - you can launch the installation wizard with the -is flag and set an alternative temporary directory. For example: SETUP.sh [-is:tempdir temporary_directory]. For additional information on disk and space requirements for the installation, refer to Chapter 2, "Supported operating systems and requirements," on page 27.
- If you are installing DB2 9.1 as part of your Tivoli Workload Scheduler installation or need to install several computers from a mounted shared directory:
  - On the installation CDs there are two types of installation scripts: a operating_system/SETUP.bin and a SETUP.sh script. The SETUP.sh wrapper script copies the product images to a local temporary directory and launches SETUP.bin. Using the SETUP.sh allows you to free your CD drive which you can then use for the DB2 images when requested to do so by the installation. After you complete the installation, you can either delete the installation images from the temporary directory or use them for other installations on the same computer or other computers where the directory is mounted.
  - On Windows systems, if you are using the DB2 product CDs instead of the images provided with Tivoli Workload Scheduler, when the DB2 installation wizard automatically starts, close it and continue the Tivoli Workload Scheduler installation.
- If the installation ends with an error, do not use the Close icon in the top right to exit the session, because this prevents the installation summary log file from being created.
- If you want to use characters of a specific language locale in the Tivoli Workload Scheduler installation path field, the language you choose for the installation wizard must match the language locale settings of the workstation you are installing.

Starting the launchpad

To access the launchpad, one of the following web browsers is required:

- Mozilla, version 1.7 and later
- Firefox, version 1.0 and later
Installing using the installation wizard

- Microsoft Internet Explorer (only for Microsoft Windows operating systems) Version 5.5 and later.

Note: On UNIX and Linux operating systems make sure you export the browser location to the BROWSER environment variable.

To start the launchpad installation program, perform the following steps:

1. Insert the CD containing the required installation image into the CD drive. For more information about locating the installation image, see “The installation CDs” on page 21.
2. The procedure for starting the launchpad varies depending on the operating system:
   - On Windows systems from the root directory of the CD run the launchpad.exe file.
   - On UNIX systems From the root directory of the CD run the launchpad.sh file.
3. The launchpad is displayed.
4. Click on Install IBM Tivoli Workload Scheduler in the left frame of the launchpad.
5. Click on one of the displayed links to install the following:
   - IBM Tivoli Workload Scheduler components (master domain manager, backup master domain manager, fault tolerant agents, command line client, connector, and language packs)
   - IBM Tivoli Dynamic Workload Console
   - IBM Tivoli Workload Scheduler Job Scheduling Console
   - IBM Tivoli Workload Scheduler Job Scheduling Console Language Pack

The related installation program starts. To proceed with the installation of the selected Tivoli Workload Scheduler component, follow the instructions described in the following sections.

The installation program prompts you for the required installation CD if necessary.

From the launchpad, you can also access information about product installation prerequisites by clicking on the different options in the left frame.

Installing a new master domain manager or backup master

When you are installing with an existing DB2 instance, DB2 must be running. If you are going to choose Oracle as the RDBMS for your Tivoli Workload Scheduler installation, the Oracle database must be running when you install.

When you are installing a master domain or backup master the process is common until you reach the RDBMS installation options. therefore, this section is divided into:

- “Common steps for installing a new master domain manager or backup master” on page 44
- “Installing a new master domain manager or backup master with an existing Oracle database” on page 47
- “Installing a new master domain manager with an existing DB2 server database” on page 48
- “Installing a new master domain manager or backup master with an existing DB2 client database” on page 50
Installing using the installation wizard

- “Installing a new master domain manager or backup master without an existing DB2 database” on page 52

Master domain manager or backup master database installation types

Before you start to install a master domain manager or backup master, you need to decide the type of database installation you are going to use. The type of database you choose directly affects the installation flow.

Table 18. Database installation flow selection

<table>
<thead>
<tr>
<th>Database type</th>
<th>Go to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installing with an existing Oracle database</td>
<td>1. “Common steps for installing a new master domain manager or backup master”</td>
</tr>
<tr>
<td></td>
<td>2. “Installing a new master domain manager or backup master with an existing Oracle database” on page 47</td>
</tr>
<tr>
<td>Installing with an existing DB2 server database</td>
<td>1. “Common steps for installing a new master domain manager or backup master”</td>
</tr>
<tr>
<td>(See Note)</td>
<td>2. “Installing a new master domain manager with an existing DB2 server database” on page 48</td>
</tr>
<tr>
<td>Installing with an existing DB2 client database</td>
<td>1. “Common steps for installing a new master domain manager or backup master”</td>
</tr>
<tr>
<td></td>
<td>2. “Installing a new master domain manager or backup master with an existing DB2 client database” on page 50</td>
</tr>
<tr>
<td>Installing a new DB2 server database (See Note)</td>
<td>1. “Common steps for installing a new master domain manager or backup master”</td>
</tr>
<tr>
<td></td>
<td>2. “Installing with a DB2 server database” on page 52</td>
</tr>
<tr>
<td>Installing a new DB2 client database</td>
<td>1. “Common steps for installing a new master domain manager or backup master”</td>
</tr>
<tr>
<td></td>
<td>2. “Installing with a DB2 client database” on page 54</td>
</tr>
</tbody>
</table>

Note: This option is available for master domain managers only. If you are installing a backup master domain manager, a local DB2 client is required. Choose the appropriate installation flow depending on whether you already have a local DB2 client or need to install a new one.

Common steps for installing a new master domain manager or backup master

When you are installing a new master domain or a backup master, you can choose to install:

- A new DB2 client or server database
- Over an existing DB2 client or server database
- Over an existing Oracle database

The initial steps are common to the three choices.
Installing using the installation wizard

To install a new instance of Tivoli Workload Scheduler, perform the following steps:

1. Insert the installation CD according to the operating system. See "The installation CDs" on page 21.
2. Run setup for the operating system on which you are installing:
   - On Windows, WINDOWS\SETUP.exe
   - On UNIX and Linux, SETUP.sh or operating_system/SETUP.bin depending on whether you want to copy the images locally.
   - Alternatively, start the launchpad as described in “Starting the launchpad” on page 42 and select the Tivoli Workload Scheduler installation.
3. Select the language of the installation wizard. Click OK.
4. Read the welcome information. Click Next.
5. Read and accept the license agreement. Click Next.
6. Select Install an Instance of Tivoli Workload Scheduler. Click Next.
7. Select Master domain manager or Backup Master domain manager as the type of Tivoli Workload Scheduler instance you are installing. Click Next.
8. Specify the Tivoli Workload Scheduler user name and password. Spaces are not permitted.
   - On Windows systems, if this user account does not already exist, it is automatically created by the installation wizard. If you specify a domain user specify the name as domain_name\user_name. If you are installing in a domain controller the user name must always be domain_name\user_name. If you specify a local user with the same name as a domain user, the local user must first be created manually by an administrator and then specified as system_name\user_name. Type and confirm the password.

   Note: The password must comply with the password policy in your Local Security Settings, otherwise, the installation fails.
   - On UNIX and Linux operating systems, this user account must be created manually before running the installation. Create a user with a home directory. By default, Tivoli Workload Scheduler is installed under the HOME directory of the selected user.

Click Next.
On Windows systems, if you specified a user account that does not already exist, you are shown information about the permissions with which the new user account will be created. Click Next.
9. Specify the following workstation configuration information:
   - Company  The name of the company.
   - This Workstation Name  The name of the workstation where you are installing the instance. The default is the hostname of the computer. The name you specify here is the name of the Tivoli Workload Scheduler workstation as it will be known in the database.
   - Master Domain Manager Name  The name of the master domain manager workstation. This field is required if you are installing a backup manager. If you are not installing a backup manager, this field is greyed out.
   - TCP/IP Port Number  The TCP/IP Port Number. The default value is 31111.
Installing using the installation wizard

**Note:** If you change this value, all default port number values in the application server port information panel are changed to reflect the new range. For example, if you specify 42111 as TCP/IP Port Number, the default for HTTP transport becomes 42125, the default for HTTPS becomes 42126 and so on.

Optionally, check the following boxes:

- **Add the default job stream "SFinal" to the Tivoli Workload Scheduler Database Definition** to add the final job stream to the database. This option allows to perform automatic production plan extension at the end of each current production plan processing. By default, this box remains unchecked. This option is available only if you are installing a Master domain manager.

- **Automatically generate all the other Tivoli Workload Scheduler Ports.** If you check this box, all the ports needed by Websphere Application Server are automatically generated using the default values and the application server port information panel is not displayed. The installation procedure checks for the availability of the ports in the specified port range. If one or more ports are in use by other applications, you are prompted to enter a new port number. By default, this box remains unchecked.

Click Next.

10. Specify the following application server port information (if you have not requested to generate ports automatically):

**HTTP Transport**
The port for the HTTP transport. The default value is 31115.

**HTTPS Transport**
The port for the secure HTTP transport. The default value is 31116.

**Bootstrap / RMI**
The port for the bootstrap or RMI. The default value is 31117.

**SOAP Connector**
The port for the application server protocol SOAP connector. The default value is 31118.

**SAS Port**
The port used by the Secure Association Services (SAS) to listen for inbound authentication requests. The default value is 31119.

**Server Authentication Port**
The port on which the Common Secure Interoperability Version 2 (CSIV2) service listens for inbound server authentication requests. The default value is 31120.

**Server Mutual Authentication Port**
The port on which the Common Secure Interoperability Version 2 (CSIV2) service listens for inbound client authentication requests. The default value is 31121.

**ORB Port**
The port used for RMI over IIOP communication. The default value is 31122.

**Administration Port**
The administrative console port. The default value is 31123.

**Administration Secure Port**
The administrative console secure port. The default value is 31124.
Installing using the installation wizard

Event Processor Monitoring Port (EIF Port)
The port used by the event management processor to receive events. The default value is 31131. This parameter is not requested if you are installing a backup master domain manager.

Note: The installation procedure checks for the availability of the ports in the specified port range. If one or more ports are in use by other applications, you are prompted to enter a new port number.

Click Next.

11. Type or Browse for the directory where the Tivoli Workload Scheduler instance is installed. On UNIX and Linux systems, the default directory is the user home directory. Click Next.

12. Select the type of RDBMS that is to be used for storing the Tivoli Workload Scheduler scheduling object definitions. Choose between Oracle and DB2 and click Next.

Depending on the type of RDBMS you are using or planning to use, refer to the following sections:

- “Installing a new master domain manager or backup master with an existing Oracle database” on page 48
- “Installing a new master domain manager with an existing DB2 server database” on page 50
- “Installing a new master domain manager or backup master with an existing DB2 client database” on page 51
- “Installing a new master domain manager or backup master without an existing DB2 database” on page 52

Installing a new master domain manager or backup master with an existing Oracle database

When you are installing with an Oracle database both for server and client, perform the following steps:

1. Specify the path of an Oracle installation that satisfies the Tivoli Workload Scheduler prerequisites. The fully-qualified path must identify a tree in the Oracle structure that includes the sqlplus executable.

   Click Next.

2. Specify the following information required to connect to the Oracle server. Retrieve this information from the Oracle database administrator.

   Net service name
   The name used by clients to identify an Oracle Net server and the specific system identifier or database for the Oracle Net connection. A net service name is mapped to a port number and protocol. Also known as a connect string, database alias, host string, or service name.

   Oracle administrator user
   The database administrator user name (such as SYSTEM) required to authenticate to the Oracle database. This account must already exist.

   Oracle administrator user password
   The database administrator user password required to authenticate to the Oracle database.

   Click Next.
Installing using the installation wizard

3. Specify the following information required to configure the Tivoli Workload Scheduler schema in the Oracle database:

**Tivoli Workload Scheduler Oracle user**
The owner of the Tivoli Workload Scheduler schema. The name must comply with the Oracle naming rules.

If you are installing a master domain manager and you leave this field blank, this name is defaulted to TwSuser. If you are installing a backup master, you are required to enter the same name used in the master domain manager.

On a fresh installation of a master domain manager this user does not exist in the database. If this is not the case, it means that there already is a master domain manager or a backup master instance pointing to the same database with this user name. If your existing Tivoli Workload Scheduler instance is version 8.3, the installation process upgrades the current database schema to the new version 8.4 schema.

If your existing instance is version 8.4, the installation process assumes that the schema is at the right level and does not create the database objects (tables, views, clusters, procedures, indexes, and so on) for Tivoli Workload Scheduler.

**Tivoli Workload Scheduler Oracle user password**
The password for the Tivoli Workload Scheduler Oracle user. It must comply with the Oracle naming rules.

**Tivoli Workload Scheduler data tablespace**
The name that identifies the Tivoli Workload Scheduler data table space. This table space must have been previously created by the database administrator. The default for this field is USERS.

**Tivoli Workload Scheduler reports tablespace**
The name that identifies the Tivoli Workload Scheduler table space where report data is to be stored. You can view the report data using the Tivoli Dynamic Workload Console.

This table space must have been previously created by the database administrator. The default value for this field is USERS.

**Tivoli Workload Scheduler temporary tablespace**
The name that identifies the Tivoli Workload Scheduler temporary table space. This table space must have been previously created by the database administrator. The default value for this field is TEMP.

Click Next.

4. Review the summary data. Click Next.

5. Click Finish.

**Installing a new master domain manager with an existing DB2 server database**

When you are installing with an existing database, perform the following steps:

1. Perform the steps described in "Common steps for installing a new master domain manager or backup master" on page 44.

2. Select Check that an existing IBM DB2 installation satisfies the Tivoli Workload Scheduler prerequisites. Click Next.

3. A DB2 discovery window is displayed. Type or Browse for the directory where the existing DB2 instance is installed. If you have more than one DB2 instance installed, make sure you provide the full path to the DB2 instance you want.
Click Next.

4. When you are installing with an existing DB2 server, specify the following DB2 instance configuration information:
   **Instance Name**
   The name of the DB2 server instance.
   **Instance Port**
   The TCP/IP port number used to communicate with the DB2 instance. The default is 50000.
   **DB2 Server administrator user**
   The user name of the administrator of the DB2 server instance. This user can also be any user having SYSADM or SYSCTRL authority on the DB2 server. On UNIX, verify that you are able to switch to this user and that it can load the DB2 environment.
   **DB2 Server administrator password**
   The password of the DB2 server administrator user, or of the user with SYSADM or SYSCTRL authority. You are asked to confirm the password.

Click Next.

5. Specify the following DB2 Enterprise Server Edition configuration information:
   **DB Name**
   The name of the DB2 database. The maximum length is 5 characters. You can use an existing DB2 database instance if its name does not exceed 5 characters. When you are installing a backup master this database must be the same as the master domain manager database. For information about DB2 database names, refer to the DB2 documentation.

Specify advanced configuration parameters for the IBM Tivoli Workload Scheduler database
   Select this option if you want to specify the following advanced parameters:
   **Tablespace Name**
   The name of the DB2 instance tablespace. This tablespace is used to store scheduling objects and event rules. For information about DB2 table spaces, refer to the DB2 documentation.
   **Tablespace Path**
   The relative path of the DB2 table space. The path can be relative or an absolute path. When the table space path is an absolute path the DB2 administrator user must have complete access rights to the directory where the table space is installed. See "DB2 tablespace relative paths" on page 57.

The default table space path name is TWS_DATA. The default table space temporary directory is TWS_TEMP. For UNIX and Linux operating systems, make sure that the DB2 Administrator has write access to the directory above the table space directory.

**Tablespace used to store reports data**
Specify the name and path of the DB2 table space where Tivoli Workload Scheduler report data is to be stored. You can view the report data using the Tivoli Dynamic Workload Console.
Installing using the installation wizard

**Report Tablespace Name**
The name of the table space for storing report data. The default name is TWS_LOG.

**Report Tablespace Path**
The path of the table space for storing report data. The default path is TWS_LOG. The path can be a relative or an absolute path. When the table space path is an absolute path the DB2 administrator user must have complete access rights to the directory where the table space is installed. See "DB2 tablespace relative paths" on page 57.

Click Next.

6. Review the summary data. Click Next.
7. Click Finish.

**Installing a new master domain manager or backup master with an existing DB2 client database**
If you are installing a backup master, you install a DB2 client that must link to the DB2 server installed on the master domain manager.

When you are installing with an existing database, perform the following steps:

1. Perform the steps described in "Common steps for installing a new master domain manager or backup master" on page 44.

2. Select Check that an existing IBM DB2 installation satisfies the Tivoli Workload Scheduler prerequisites. Click Next.

3. A DB2 discovery window is displayed. Type or Browse for the directory where the existing DB2 instance is installed. If you have more than one DB2 instance installed, make sure you provide the full path to the DB2 instance you want. Click Next.

4. Specify the following DB2 instance configuration information:
   **Remote Database Server**
   The IP address or host name of the computer where the DB2 server is installed.

   **Remote Database Port**
   The TCP/IP port number the remote DB2 server instance uses to communicate.

**Identify the user on the remote DB2 server to be used by the installation for DB2 administration tasks**
Provide the following data:

   **DB2 Server administrator user**
   The user name of the administrator of the DB2 server instance. This user can also be any user having SYSADM or SYSCTRL authority on the DB2 server. On UNIX, verify that you are able to switch to this user and that it can load the DB2 environment.

   **DB2 Server administrator password**
   The password of the DB2 server administrator user, or of the user with SYSADM or SYSCTRL authority. You are asked to confirm the password.

**Identify the user on the DB2 client to be used by the installation for DB2 administration tasks**
Specify the user on the DB2 client to be used by the installation for DB2 administration tasks. Provide the following data:
DB2 Client Administrator User
The user name of the DB2 administrator of the DB2 client instance.

DB2 Client Administrator Password
The password of the DB2 administrator of the DB2 client instance.

Note: The password must comply with the password policy in your Local Security Settings otherwise, the installation fails.

Identify the user on the DB2 server to be used by Tivoli Workload Scheduler to access the database, if different from the DB2 Server Administration User
Select this option when the DB2 server user used to access Tivoli Workload Scheduler is different from the DB2 Server Administration User. Provide the following data:

Tivoli Workload Scheduler DB2 User
The user name of the Tivoli Workload Scheduler DB2 user.

Tivoli Workload Scheduler DB2 Password
The password of the Tivoli Workload Scheduler DB2 user.

For information about DB2 configuration data, refer to the DB2 documentation. Click Next.

5. Specify the following DB2 Enterprise Server Edition configuration information:

DB Name
The name of the DB2 database. The maximum length is 5 characters. You can use an existing DB2 database instance as long as its name does not exceed 5 characters. When you are installing a backup master this database must be the same as the master domain manager database. For information about DB2 database names, refer to the DB2 documentation.

Specify advanced configuration parameters for the IBM Tivoli Workload Scheduler database
Select this option if you want to specify the following advanced parameters:

Tablespace Name
The name of the DB2 instance table space. For information about DB2 table spaces, refer to the DB2 documentation.

Tablespace Path
The relative path of the DB2 table space. The path can be a relative or an absolute path. When the table space path is an absolute path the DB2 administrator user must have complete access rights to the directory where the table space is installed. See “DB2 tablespace relative paths” on page 57.

The default table space path name is TWS_DATA. The default table space temporary directory is TWS_TEMP. For UNIX and Linux operating systems, make sure that the DB2 Administrator has write access to the directory above the table space directory.

Tablespace used to store reports data
Specify the name and path of the DB2 table space where Tivoli
Installing using the installation wizard

Workload Scheduler report data is to be stored. You can view report data using the Tivoli Dynamic Workload Console.

**Report Tablespace Name**

The name of the table space for storing report data. The default name is TWS_LOG.

**Report Tablespace Path**

The path of the table space for storing report data. The default path is TWS_LOG. The path can be relative or an absolute path. When the table space path is an absolute path the DB2 administrator user must have complete access rights to the directory where the table space is installed. See "DB2 tablespace relative paths" on page 57.

Click Next.

6. Review the summary data. Click Next.

7. Click Finish.

**Installing a new master domain manager or backup master without an existing DB2 database**

When you are installing a new instance without an existing installation of DB2, choose whether you are:

- "Installing with a DB2 server database" *(option available only for master domain managers)*
- "Installing with a DB2 client database" on page 54

When you are installing a DB2 database to a UNIX or Linux operating system, use the SETUP.sh script that is located at the root of the relevant CD. See "The installation CDs" on page 21. The wrapper script moves the installation files to a local folder and launches the installation from there. This frees the CD-ROM drive thereby allowing you to change to the DB2 CD when requested to do so by the installation. Alternatively, you can copy the DB2 files to a hard drive and run SETUP.bin from there.

You launch the wrapper script using the following syntax:

/\mnt/cdrom/SETUP.sh [-is:tempdir local directory] [InstallShieldParam]

where:

/\mnt/cdrom
Represents where the CD is mounted.

local directory
Is the local temporary directory where the installation image is copied.

InstallShieldParam
Represents any other InstallShield installation parameters used.

For example:

/\mnt/cdrom/SETUP.sh -is:tempdir /\images -is:javaconsole -is:log /\tmp/install.log

**Installing with a DB2 server database:** When you are installing a DB2 server database, perform the following steps:

1. Follow the steps in "Common steps for installing a new master domain manager or backup master" on page 44.
3. Specify the following DB2 V9.1 Enterprise Server information:
   **Instance Name**
   - The name of the DB2 server instance.
   **Instance Port**
   - The TCP/IP port number used to communicate with the DB2 instance. The default is 50000.
   **DB2 Server Administrator User**
   - The user name of the administrator of the DB2 server instance.
   **DB2 Server Administrator Password**
   - The password of the administrator of the DB2 server instance.

   **Note:** The password must comply with the password policy in your Local Security Settings otherwise, the installation fails.

   **Confirm DB2 Server Administrator Password**
   - Confirm the password.

   Click Next.
4. Specify the DB2 Universal Database™ Enterprise Server installation directory.
   Click Next.
5. Specify the following DB2 V9.1 Enterprise Server configuration information:
   **DB Name**
   - The name of the DB2 database. The maximum length is 5 characters.
   You can use an existing DB2 database instance as long as its name does not exceed 5 characters. When you are installing a backup master this database must be the same as the master domain manager database.

   **Specify advanced configuration parameters for the IBM Tivoli Workload Scheduler database**
   - Select this option if you want to specify the following advanced parameters:

   **Tablespace Name**
   - The name of the DB2 tablespace used to store scheduling objects and event rules. For information about DB2 tablespaces, refer to the DB2 documentation.

   **Tablespace Path**
   - The path of the DB2 tablespace. The path can be relative or absolute. When the tablespace path specified is an absolute path the DB2 administrator user must have complete access rights to the directory where the tablespace is installed. See "DB2 tablespace relative paths" on page 57.

   The default tablespace path name is TWS_DATA. The default tablespace temporary directory is TWS_TEMP. For UNIX and Linux operating systems, make sure that the DB2 Administrator has write access to the directory above the tablespace directory.

   **Tablespace used to store reports data**
   - Specify the name and path of the DB2 table space where Tivoli Workload Scheduler report data is to be stored. You can view the report data using the Tivoli Dynamic Workload Console.

   **Report Tablespace Name**
   - The name of the table space for storing report data. The default name is TWS_LOG.
Installing using the installation wizard

Report Tablespace Path

The path of the table space for storing report data. The default path is TWS_LOG. The path can be a relative or an absolute path. When the table space path is an absolute path the DB2 administrator user must have complete access rights to the directory where the table space is installed. See “DB2 tablespace relative paths” on page 57.

Click Next.

If the DB2 Server Administration user specified in step 3 on page 53 does not exist, an information panel is displayed informing you that it will be created. Click Next.

6. Review the summary data. Click Next.
7. Enter the path where the DB2 images are located.
8. Review the summary data. Click Open.

When the installation ends, click Finish.

Installing with a DB2 client database: When you are installing a DB2 client database, perform the following steps:

1. Follow the steps in “Common steps for installing a new master domain manager or backup master” on page 44.
2. Select Install DB2 UDB Administration Client, version 9.1. Click Next.
3. Specify the following DB2 V9.1 Administration Client information:
   - Remote Database Server
     The IP address or host name of the computer where the DB2 server is installed.
   - Remote Database Port
     The TCP/IP port number the remote DB2 server instance uses to communicate.

Identify the user on the remote DB2 server to be used by the installation for DB2 administration tasks

Specify the DB2 Server Administrator User. Provide the following data:

   - DB2 Server Administrator User
     The administrator user of the DB2 server instance.
   - DB2 Server Administrator Password
     The administrator user password of the DB2 server instance.

Identify the user on the DB2 client to be used by the installation for DB2 administration tasks

Specify the user on the DB2 client to be used by the installation for DB2 administration tasks. Provide the following data:

   - DB2 Client Administrator User
     The user name of the DB2 administrator of the DB2 client instance.
   - DB2 Client Administrator Password
     The password of the DB2 administrator of the DB2 client instance.

Note: The password must comply with the password policy in your Local Security Settings otherwise, the installation fails.
Identify the user on the DB2 server to be used by Tivoli Workload Scheduler to access the database, if different from the DB2 Server Administration User

Specify the user on the DB2 server, if different from the DB2 Server Administration User. Provide the following information:

**Tivoli Workload Scheduler DB2 User**

The user name of the Tivoli Workload Scheduler DB2 user.

**Tivoli Workload Scheduler DB2 Password**

The password of the Tivoli Workload Scheduler DB2 user.

4. Click Next.

5. Specify the following DB2 V9.1 Administration Client configuration information:

**DB Name**

The name of the DB2 database. The maximum length is 5 characters. You can use an existing DB2 database instance as long as its name does not exceed 5 characters. When you are installing a backup master this database must be the same as the master domain manager database.

**Specify advanced configuration parameters for the IBM Tivoli Workload Scheduler database**

Select this option if you want to specify the following advanced parameters:

**Tablespace Name**

The name of the DB2 instance tablespace. For information about DB2 tablespaces, refer to the DB2 documentation.

**Tablespace Path**

The relative path of the DB2 tablespace. The path can be relative or an absolute path. When the tablespace path is an absolute path the DB2 administrator user must have complete access rights to the directory where the tablespace is installed. See "DB2 tablespace relative paths" on page 57.

The default tablespace path name is TWS_DATA. The default temporary directory is TWS_TEMP. For UNIX and Linux operating systems, make sure that the DB2 Administrator has write access to the directory above the tablespace directory.

**Tablespace used to store reports data**

Specify the name and path of the DB2 table space Tivoli Workload Scheduler where Tivoli Workload Scheduler report data is to be stored. The report data can be viewed with the Tivoli Dynamic Workload Console.

**Tablespace Name**

The name of the table space for storing report data. The default name is TWS_LOG.

**Tablespace Path**

The path of the table space for storing report data. The default path is TWS_LOG.

Click Next.

6. Enter the path where the DB2 images are located.

7. Review the summary data. Click **Finish**.
Installing using the installation wizard

Installing a new agent or domain manager

To install a new agent or domain manager of Tivoli Workload Scheduler, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.
2. Run setup for the operating system on which you are installing:
   - On Windows operating systems, WINDOWS\SETUP.exe
   - On UNIX and Linux operating systems, operating_system/SETUP.bin
3. Select the language of the installation wizard. Click OK.
4. Read the welcome information. Click Next.
5. Read and accept the license agreement. Click Next.
6. Select Install an Instance of Tivoli Workload Scheduler. Click Next.
7. Select Agent or Domain Manager. Click Next.
8. Specify the Tivoli Workload Scheduler user name and password. Spaces are not permitted.
   - On Windows systems, if this user account does not already exist, it is automatically created by the installation wizard. If you specify a domain user specify the name as domain_name\user_name. If you are installing in a domain controller the user name must be always domain_name\user_name. If you specify a local user with the same name as a domain user, the local user must first be created manually by an administrator and then specified as system_name\user_name.

   Note: The password must comply with the password policy in your Local Security Settings otherwise, the installation fails.
   - On UNIX and Linux operating systems, this user account must be created manually before running installation. Create a user with a home directory. Tivoli Workload Scheduler is installed by default under the home directory of the selected user.

   Click Next.
9. On Windows systems, if you specified a user name that does not already exist, an information panel opens. Review the information and click Next.
10. Specify the following workstation configuration information:

    Company  The name of the company.

    This Workstation Name  The name of the workstation where you are installing the instance. The default is the hostname of the computer. The name you specify here is the name of the Tivoli Workload Scheduler workstation as it will be known in the database.

    Master Domain Manager Name  The name of the master domain manager to which the workstation belongs.

    TCP/IP Port Number  The TCP/IP Port Number. The default value is 31111.

   Click Next.
11. Type or Browse for the directory where the Tivoli Workload Scheduler instance is installed.
12. Review the summary data. Click Next.
13. Click Finish.

**DB2 tablespace relative paths**

When you create a DB2 tablespace with a relative path, the path is constructed in the following way:

\[ DFTDBPATH\DB2_instance\NODE0000\SQLnnnnn\TABLESPACE_REL_PATH \]

where:

- **DFTDBPATH**
  - For UNIX and Linux operating systems, this is the home instance of the DB2 installation. For Windows operating systems, this is the drive where the DB2 instance is installed.

- **DB2_instance**
  - Is the DB2 instance.

- **NODE0000**
  - Is the directory where DB2 database instances are located.

- **SQLnnnnn**
  - Is an incremental directory path that depends on the number of database instances.

- **TABLESPACE_REL_PATH**
  - Is the relative path you specified for the tablespace.

For more information about tablespace relative paths, refer to the DB2 documentation set.

**Installing a command-line client**

The command-line client is a component of Tivoli Workload Scheduler that implements many of the commands used on the master domain manager. It can be installed on any workstation that satisfies its prerequisites, including workstations where no other Tivoli Workload Scheduler components are installed. It communicates by TCP/IP with the command-line server, which is part of the master domain manager. The command-line client is installed using the InstallShield installation wizard, interactive or silent.

The information required to make the connection with the master domain manager must be defined either in the local options file or supplied as parameters to the command.

The commands you can use are the following:

- Composer
- Optman
- Planman `showinfo` and `unlock` (the other planman commands need to be run locally on the master domain manager)

**Note:** The command-line client is different from and independent from, the ability to use `conman` locally on an agent to manage the local Symphony file and local jobs.

**Installing a command-line client**

To install a command-line client on an existing installation, perform the following steps:
Installing using the installation wizard

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.

2. Run setup for the operating system on which you are installing:
   - On Windows, WINDOWS\SETUP.exe
   - On UNIX and Linux, SETUP.bin
   - Alternatively, start the launchpad as described in “Starting the launchpad” on page 42.

3. Select the installation wizard language. Click OK.
4. Read the welcome information. Click Next.
5. Read and accept the license agreement. Click Next.
7. Specify the following information:

   **Remote Host**
   The TCP/IP address or host name of the computer where the Tivoli Workload Scheduler engine is installed.

   **Remote Port**
   The HTTP or HTTPS port number used to connect to the computer where the master domain manager is installed. This port number must match the values defined for the master domain manager.

   **User Name**
   The user name used to connect to the computer where the master domain manager is installed. This user should be a valid user listed in the security file on the master domain manager.

   **Password**
   The password used to connect to the computer where the master domain manager is installed.

   Click Next.
8. Select the installation directory. Click Next.
9. Review the installation settings. Click Next.
10. When the installation completes, a panel displays a successful installation or indicates the location of the log file if the installation was unsuccessful. Click Finish.

Adding a new feature

You can install the following optional components or features that were not installed during a previous installation:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tivoli Workload Scheduler connector</td>
<td>The Job Scheduling Console communicates with the Tivoli Workload Scheduler system through the connector. It translates instructions entered through the console into scheduler commands.</td>
</tr>
<tr>
<td>Language Packs</td>
<td>The English language pack and the language locale of the operating system are installed by default. You can also install any of the supported languages.</td>
</tr>
</tbody>
</table>
Adding a connector

To add a connector instance to an existing installation, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.
2. Run setup for the operating system on which you are installing:
   - On Windows, WINDOWS\SETUP.exe
   - On UNIX and Linux, SETUP.bin
   - Alternatively, start the launchpad as described in “Starting the launchpad” on page 42.
3. Select the installation wizard language. Click OK.
4. Read the welcome information. Click Next.
5. Read and accept the license agreement. Click Next.
6. From the drop down list, select an existing installation. Existing installations are identified by the type of agent and the user name for which the agent was installed.
7. Select Add a New Feature. Click Next.
8. Select Tivoli Workload Scheduler Connector, V8.4. Click Next.
9. Type the password of the TWSuser. Click Next.
10. Specify the following application server port information:
    - **HTTP Transport**
      The port for the HTTP transport. The default value is 31115.
    - **HTTPS Transport**
      The port for the secure HTTP transport. The default value is 31116.
    - **Bootstrap / RMI**
      The port for the bootstrap or RMI. The default value is 31117.
    - **SOAP Connector**
      The port for the application server protocol SOAP connector. The default value is 31118.
    - **SAS Port**
      The port used by the Secure Association Services (SAS) to listen for inbound authentication requests. The default value is 31119.
    - **Server Authentication Port**
      The port on which the Common Secure Interoperability Version 2 (CSIV2) service listens for inbound server authentication requests. The default value is 31120.
    - **Server Mutual Authentication Port**
      The port on which the Common Secure Interoperability Version 2 (CSIV2) service listens for inbound client authentication requests. The default value is 31121.
    - **ORB Port**
      The port used for RMI over IIOP communication. The default value is 31122.
    - **Administration Port**
      The administrative console port. The default value is 31123.
    - **Administration Secure Port**
      The administrative console secure port. The default value is 31124.
Installing using the installation wizard

**Note:** The installation procedure checks for the availability of the ports in the specified port range. If one or more ports are in use by other applications, you are prompted to enter a new port number.

Click Next.

11. Review the installation settings. Click Next.

12. When the installation completes, a panel displays a successful installation or indicates the location of the log file if the installation was unsuccessful. Click Finish.

Adding language packs to a command-line client

To add language packs to a command-line client, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CD” on page 21.

2. Run setup for the operating system on which you are installing:
   - On Windows, WINDOWS\SETUP.exe
   - On UNIX and Linux, SETUP.bin

3. Select the installation wizard language. Click OK.

4. Read the welcome information. Click Next.

5. Read and accept the license agreement. Click Next.

6. From the drop-down list, select an existing command-line client. Existing installations are identified by the type and the user name, such as Tivoli Workload Scheduler 8.4 another\another [CLI].

7. Select Add a New Feature. Click Next.

8. Select the additional languages to install. Click Next.

9. Review the installation settings. Click Next.

10. When the installation completes, a panel displays a successful installation or indicates the location of the log file if the installation was unsuccessful. Click Finish.

Performing a silent installation

Use the response file templates provided on the CDs in the \RESPONSEFILES\ directory to perform a silent installation. See “The installation CD” on page 21.

The files include all the information required by the installation program to run without user intervention. Instructions for customizing the files are included in the files as commented text.

Table 20 lists the response files available and the type of installation each performs:

<table>
<thead>
<tr>
<th>Type of installation</th>
<th>Response file to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command-line client</td>
<td>TWS84_CLI.txt</td>
</tr>
<tr>
<td>Fresh agent</td>
<td>TWS84_FRESH_Agent.txt</td>
</tr>
<tr>
<td>Fresh connector</td>
<td>TWS84_FRESH_Connector.txt</td>
</tr>
<tr>
<td>Fresh master domain manager</td>
<td>TWS84_FRESH_MDM.txt</td>
</tr>
<tr>
<td>Fresh backup master domain</td>
<td>TWS84_FRESH_BACKUP_MDM.txt</td>
</tr>
<tr>
<td>Command line client upgrade</td>
<td>TWS84_UPGRADE_CLI.txt</td>
</tr>
<tr>
<td>Agent upgrade</td>
<td>TWS84_UPGRADE_Agent.txt</td>
</tr>
</tbody>
</table>
Table 20. Response files (continued)

<table>
<thead>
<tr>
<th>Type of installation</th>
<th>Response file to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connector upgrade</td>
<td>TWS84_UPGRADE_Connector.txt</td>
</tr>
<tr>
<td>Master domain manager upgrade</td>
<td>TWS84_UPGRADE_MDM.txt</td>
</tr>
<tr>
<td>Backup master domain upgrade</td>
<td>TWS84_UPGRADE_BACKUP_MDM.txt</td>
</tr>
<tr>
<td>Agent uninstall</td>
<td>TWS84_UNINSTALL_Agent.txt</td>
</tr>
</tbody>
</table>

**Note:** When you are performing a silent installation on UNIX zSeries systems, you must first save the response file in UTF 8 format.

**Silent installation procedure**

For a silent installation, perform the following steps:

1. Copy the relevant response file to a local directory and edit it to meet the needs of your environment.
2. Save the file with your changes.
3. Enter the following command:
   - On Windows operating systems,
     ```bash```
     ```
     SETUP.exe -options <local_dir>\response_file.txt -silent
     ```
   - On UNIX and Linux operating systems,
     ```bash```
     ```
     ./SETUP.bin -options <local_dir>/response_file.txt -silent
     ```

   The SETUP.exe file is located in the WINDOWS directory. See "The installation CDs" on page 21.

4. Review the installation messages in the summary.log file to check that installation was successful.
5. At the end of a successful installation, perform one of the following configuration tasks depending on the type of agent you installed:
   - ”Configuring a master domain manager” on page 140
   - ”Configuring an agent” on page 141

---

## Installing Tivoli Workload Scheduler agents using twsinst

This section explains the command line method of installing using the **twsinst** script. It contains the following subsections:

* Installing a new agent instance

---

### Installing a new agent instance

You can use the **twsinst** script to install agents on UNIX and Linux

Refer to "Supported operating systems” on page 27 for a complete list of supported operating systems.

During the installation process, **twsinst** creates a file in the TwShome directory for each of the installation steps. If you stop and restart the installation, the installation process starts from the installation step where it was stopped.
Installing agents using twsinst

To install a Tivoli Workload Scheduler agent instance, perform the following steps:

1. Insert the CD according to the operating system. See “The installation CDs” on page 21.

2. Create the Tivoli Workload Scheduler user. The software is installed by default in the user’s home directory, referred to as TWShome.

   User:  TWSuser

   Home:  TWShome (for example: /opt/TWS)

3. Log in as root and on the operating system where you want to install.

4. Run the twsinst script.

A successful installation using the twsinst issues the return code RC = 0. A failed installation issues the return code RC = 1. In the case of a failed installation, refer to IBM Tivoli Workload Scheduler Administration and Troubleshooting.

Synopsis

Show command usage and version

twsinst -u | -v

Install a new instance

twsinst -new -uname <username>
   [-thiscpu <cpuname>]
   [-master <cpuname>]
   [-port <port_number>]
   [-company <company_name>]
   [-inst_dir <install_dir>]
   [-lang <lang_id>]
   [-create_link]
   [-skip_usercheck]
   [-reset_perm]

Parameters

- u    Displays command usage information and exits.
- v    Displays the command version and exits.
- new  Specifies the type of installation to perform:

- new  A fresh installation of Tivoli Workload Scheduler, version 8.4.

   Installs an agent or master and all supported language packs.

- uname <username>

   The name of the user for which Tivoli Workload Scheduler is installed, updated, or uninstalled. This user name is not to be confused with the user performing the installation logged on as root. For a new installation, this user account must be created manually before running the installation. Create a user with a home directory. Tivoli Workload Scheduler is installed by default under the HOME directory of the specified user.

- thiscpu <cpuname>

   The name of the Tivoli Workload Scheduler workstation of this installation. The name cannot exceed 16 characters. This name is registered in the localopts file. If not specified, the default value is the hostname of the workstation.

- master <cpuname>

   The workstation name of the master domain manager. This name cannot exceed 16 characters and cannot contain spaces. If not specified, the default value is MASTER.
Installing agents using twsinst

-port <port_number>
The TCP/IP port number. This number is registered in the localopts file. The default value is 31111.

-company <company_name>
The name of the company. The company name cannot contain blank characters. The name appears in program headers and reports. If not specified, the default name is COMPANY.

-inst_dir <install_dir>
The directory of the Tivoli Workload Scheduler installation. This path cannot contain blanks. If not specified, the path is set to the username home directory.

-lang <lang_id>
The language in which the twsinst messages are displayed. If not specified, the system LANG is used. If the related catalog is missing, the default C language catalog is used.

Note: This is the language in which the installation log is recorded, and not the language of the installed engine instance. twsinst installs all languages as default.

-create_link
Create the symlink. See Table 9 on page 23 for more information.

-skip_usercheck
Skip the check of the user in the /etc/password file or using the su command. Enable this option if the authentication process within your organization is not standard, thereby disabling the default authentication option.

-reset_perm
Reset the permissions of the libatrc library.

Examples
For example, a sample twsinst script for installing a new instance of a fault-tolerant agent workstation:

./twsinst -new -uname twsuser -thistcpu phughes -master TWSmdm -port 37124 -company IBM

Installing Tivoli Workload Scheduler agents using Software Distribution

This section describes how to install using Software Distribution software package blocks.

Software packages and parameters
Tivoli Workload Scheduler can be installed distributing a software package block (SPB), using the Software Distribution component of Tivoli Configuration Manager, Versions 4.1, 4.2, 4.2.1, 4.2.2, or 4.2.3. You can distribute the SPB, using either the command line interface or from the Tivoli desktop.

Note: Do not modify the SPB supplied with the product.

An SPB exists for each supported operating system located on the installation disks under the directory of the operating system. The SPBs are named according to the operating system: Tivoli_TWS_<operating_system>.SPB. In order for the packages
Software Distribution installation

to be distributed, they must be imported in software package profiles. The software package profiles must be named according to the operating system and user: FP_TWS_<operating_system>_<twsuser>.8.4.00. Possible values for operating system are:

- AIX
- HP
- SOLARIS
- WINDOWS
- LINUX_I386
- LINUX_PPC
- LINUX_S390
- SOLARIS_I386
- HPIA64
- LINUX_X86_64
- WINDOWS_X86_64

An SPB also exists to install the language packs that is generic for all languages: Tivoli_TWS_LP.SPB. The software package profiles must be named according to the user: Tivoli_TWS_LP_<twsuser>.8.4.00. The language pack software package block is found under the root directory of the installation CDs.

A number of Tivoli Workload Scheduler parameters are used by the software package block to perform the installation. These parameters are defined as default variables in the software package. Table 21 is a list of installation parameters.

Table 21. SPB installation parameters

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>installer</td>
<td>This parameter is for Windows operating systems only. It is the user ID of the installer of Tivoli Workload Scheduler. The default value is Administrator.</td>
</tr>
<tr>
<td>install_dir</td>
<td>This required parameter is the fully qualified path to the location of the Tivoli Workload Scheduler installation. This path cannot contain blanks. On Windows workstations, this path is created if it does not already exist. On UNIX and Linux operating systems, this path is the same as the user’s home directory. The default values are:</td>
</tr>
<tr>
<td></td>
<td>• Windows: ${system_drive}\win32appTWS${tws_user}</td>
</tr>
<tr>
<td></td>
<td>• UNIX and Linux: opt/TWS/${tws_user}</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>tws_user</td>
<td>This required parameter is the user name for which Tivoli Workload Scheduler instance is being installed. On Windows systems, if this user account does not already exist, it is automatically created by the installation. If you specify a domain user or domain controller, you need to specify the domain in the domain variable. If you specify a local user with the same name as a domain user, the local user must first be created manually by an Administrator and then identified as system_name\user_name. On UNIX and Linux operating systems, this user account must be created manually before running the installation. The default value is ${user_name}.</td>
</tr>
<tr>
<td>domain</td>
<td>This parameter is for Windows operating systems only. This is an optional parameter unless the user is a domain user. The domain name of the user. If you specify a domain user, specify the name as domain_name\user_name. The default value is ${computer_name}.</td>
</tr>
<tr>
<td>backup_dir</td>
<td>This optional parameter is when you are performing an upgrade. It indicates the location to where the current installation is copied before it is upgraded. The default value is ${install_dir}<em>backup</em>${tws_user}.</td>
</tr>
<tr>
<td>pwd (for Windows only)</td>
<td>This is a required parameter for Windows operating systems when performing a first time install. The password associated with the tws_user user name. The SPB password variable is passed to the pwd variable.</td>
</tr>
<tr>
<td>ft_agent</td>
<td>This parameter must always be set to true.</td>
</tr>
<tr>
<td>company</td>
<td>This optional parameter is the company name. This name appears in program headers and reports. The default value is COMPANY.</td>
</tr>
<tr>
<td>this_cpu</td>
<td>This required parameter is the name of the workstation on which you are performing the installation. This name cannot exceed 16 characters and cannot contain spaces. The default value is THIS_CPU.</td>
</tr>
<tr>
<td>master_cpu</td>
<td>This optional parameter is the name of the master domain manager. This name cannot exceed 16 characters and cannot contain spaces. The default is MASTER.</td>
</tr>
<tr>
<td>tcp_port</td>
<td>This required parameter is the TCP/IP port number used by the instance being installed. When installing more than one instance on the same workstation, use different port numbers for each instance. It must be an unassigned 16-bit value in the range 1-65535. The default value is 31111.</td>
</tr>
<tr>
<td>fresh_install</td>
<td>This required parameter indicates whether this is a first time install. Specify true to perform a fresh installation. Specify false to perform an upgrade. The default value is true.</td>
</tr>
</tbody>
</table>
Software Distribution installation

Table 21. SPB installation parameters (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>upgrade</td>
<td>This required parameter indicates whether the install is an upgrade. Specify false to perform a fresh installation. Specify true to perform an upgrade. The default value is false.</td>
</tr>
<tr>
<td>from_release</td>
<td>When you specify upgrade=&quot;true&quot; you must also specify from_release indicating the release of the existing instance. The default value is 8.2 and the format is 8.x.x.</td>
</tr>
<tr>
<td>backup</td>
<td>This optional parameter indicates a backup. Specify false for a fresh install. The default value is false.</td>
</tr>
</tbody>
</table>

Notes:
1. fresh_install and upgrade are mutually exclusive.
2. Those variables that are not documented here, are for debugging purposes only. See IBM Tivoli Workload Scheduler Administration and Troubleshooting.

Installation procedure

When installing, the installation procedure checks that there is sufficient space for the Tivoli Workload Scheduler engine to be installed. It does not, however, check that there is sufficient space for the Configuration Manager backup directory specified in the swdis.ini file. Before you begin installing, make sure that there is enough space available in the directory specified in the backup_dir parameter in the swdis.ini file according to the operating system. Refer to the Tivoli Configuration Manager documentation for these space requirements.

To perform the installation, complete the following steps:
1. Create a software package profile:
   FP_TWS_<operating_system>_TWSuser.8.4.00 where operating_system is the operating system where you are installing and TWSUser is the user of the installation.
2. Import the software package block using the wimspso command. When you import the software package block, you must pass the name of the profile to wimspso so that the Configuration Manager endpoint catalogs the name correctly.
3. Install the software package block using the winstsp command.
   Note that the supplied software packages must be installed as COMMITTED. The packages cannot be installed as UNDOABLE because the UNDO action does not rollback the product registry entries.
4. Perform one of the following configuration tasks depending on the type of agent you installed:
   • “Configuring an agent” on page 141

Note:
Syntax
The following is an example of the settings required to perform a fresh installation of an agent on a UNIX workstation, with the user juno and the endpoint text_ESP. See Table 21 on page 64 for a description of the parameters.

```
winstsp -D install_dir="/home/TWS/juno"
-D tws_user="juno" -D ft_agent="true" -D this_cpu="saturn" -D tcp_port="31111"
-D master_cpu="saturn_2" -D fresh_install="true" -D upgrade="false"
-D backup="false" FP_TWS_WINDOWS_juno.spb test_EP
```

Installing language packs
You can also install language packs using Software Distribution. Locate the Tivoli_TWS_LP.SPB software package block in the root directory of CD1, and then customize the following parameters before you install.

<table>
<thead>
<tr>
<th>Default variable</th>
<th>Description</th>
<th>Required</th>
<th>Default value</th>
</tr>
</thead>
<tbody>
<tr>
<td>zh_CN</td>
<td>Chinese, Simplified</td>
<td></td>
<td>Specify true for the languages to install. The default value for all other languages is false.</td>
</tr>
<tr>
<td>it</td>
<td>Italian</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ko</td>
<td>Korean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>es</td>
<td>Spanish</td>
<td></td>
<td></td>
</tr>
<tr>
<td>zh_TW</td>
<td>Chinese, Traditional</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ja</td>
<td>Japanese</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pt_BR</td>
<td>Brazilian Portuguese</td>
<td></td>
<td></td>
</tr>
<tr>
<td>de</td>
<td>German</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fr</td>
<td>French</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALL_LANG</td>
<td>All of the above languages.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tws_user</td>
<td>The user name for which the specified language pack is being installed.</td>
<td>Yes</td>
<td>$(user_name)</td>
</tr>
<tr>
<td>install_dir</td>
<td>The fully qualified path to which the specified language packs are installed.</td>
<td>Yes</td>
<td>$(program_files)</td>
</tr>
</tbody>
</table>

The following is the syntax required to install all languages:

```
winstsp -D install_dir="Installation Path" -D tws_user="UserName"
[D zh_C=true ... -D de=true | ALL_LANG=true] Tivoli_TWS_LP.SPB [subscribers...]```

The following is the syntax required to install Italian and German language packs:

```
winstsp -D install_dir="Installation Path" -D tws_user="UserName"
[D it=true | -D de=true] Tivoli_TWS_LP.SPB [subscribers...]```
Software Distribution installation
Chapter 4. Using the Tivoli Workload Scheduler tutorial utility

This chapter describes the Tivoli Workload Scheduler tutorial utility and guides you through a set of steps to populate and use a standalone test environment. This tutorial utility is intended for first time users of Tivoli Workload Scheduler who want an overview of the features and capabilities of the product in a real environment. The tutorial utility includes a sample database with predefined scheduling objects and a set of scenarios that use these objects.

The `sampledbsetup.sh` or the `SAMPLEDBSETUP.CMD` script (depending on whether you are in a UNIX or Windows environment) populates your Tivoli Workload Scheduler with a set of scheduling objects. The scenario scripts use these objects in basic scheduling activities. Each scenario is self-contained and can be run in any order, with the exception of the first scenario which is a prerequisite to all others.

The Tivoli Workload Scheduler tutorial utility runs only on a master domain manager. It does not affect any other workstation defined in your Tivoli Workload Scheduler environment. Each scenario is launched as a separate script file which uses the `conman` and `composer` command interfaces. The syntax and usage of each command covered in the scenarios is explained in detail in the *Tivoli Workload Scheduler: Reference Guide*. Before you begin using the utility, read an overview of Tivoli Workload Scheduler concepts and tasks in *Tivoli Workload Scheduler Suite* Tivoli Workload Scheduling Suite: Overview.

This chapter is divided into the following sections:

- “Populating your Tivoli Workload Scheduler database”
- “Overview of the scheduling scenarios” on page 70
- “Creating and working with the production plan” on page 71
- “Running the scheduling scenarios” on page 72
- “Removing the tutorial objects from the database” on page 75

Populating your Tivoli Workload Scheduler database

This section describes how to use the utility to populate your Tivoli Workload Scheduler database.

After you have installed Tivoli Workload Scheduler on the master domain manager in your test environment you are ready to populate the database.

Follow these steps:

1. Go to the `<TWS_home>/TWStutorial` directory, where `TWS_home` is the home directory of the user for which you have installed Tivoli Workload Scheduler.
2. Launch the tutorial utility installation script:

   In a Windows environment:
   - `SAMPLEDBSETUP.CMD`

   In a UNIX environment:
   - `sampledbsetup.sh`
Populating the database

The script adds a set of scheduling objects with names starting with the letters SMPL, followed by the object type and scenario number so that all objects used in each scenario are easily identifiable. Some objects are different depending on whether you are using a UNIX or a Windows environment.

The script performs a check on the database. If any objects with the same name are found, you are prompted to specify if these objects can be overwritten.

When processing of the script ends successfully, your Tivoli Workload Scheduler database contains the objects needed to run the scheduling scenarios.

Objects used by the Tivoli Workload Scheduler tutorial scenarios

After you have successfully installed the Tivoli Workload Scheduler tutorial utility in your test environment, your database is populated with the following scheduling objects:

Table 23. Objects downloaded by the tutorial utility

<table>
<thead>
<tr>
<th>Object type</th>
<th>Object Names (Total objects)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calendar</td>
<td>SMPCAL6 (1)</td>
</tr>
<tr>
<td>Parameter</td>
<td>SMPLHOME, SMPLUSER, SMPLWin1 to SMPLWin4 or SMPLUnx1 to SMPLUnx4 (6)</td>
</tr>
<tr>
<td>Resource</td>
<td>SMPLRES1, SMPLRES2 (2)</td>
</tr>
<tr>
<td>Prompt</td>
<td>SMPLPRM4, SMPLPRM5, SMPLPRM6, SMPLPRM7 (4)</td>
</tr>
<tr>
<td>Job</td>
<td>SMPL_JOB_3_0_1, SMPL_JOB_3_0_2, SMPL_JOB_3_0_3, SMPL_JOB_4_0_1, SMPL_JOB_4_0_3, SMPL_JOB_5_0_1, SMPL_JOB_5_0_2, SMPL_JOB_EVN, SMPL_JOB_ODD, SMPL_JOB_SBJ, SMPL_JOB_7_0_LAST, SMPL_JOB_7_0_RECV, SMPL_JOB_7_0_1, SMPL_JOB_7_0_2, SMPL_JOB_7_0_3 (16)</td>
</tr>
<tr>
<td>Job Stream</td>
<td>SMPL_SCHED_3_0_1, SMPL_SCHED_3_0_2, SMPL_SCHED_4_0_1, SMPL_SCHED_4_0_2, SMPL_SCHED_4_0_S, SMPL_SCHED_5_0_1, SMPL_SCHED_5_0_2, SMPL_SCHED_5_ODD, SMPL_SCHED_5_EVN, SMPL_SCHED_5BS, SMPL_SCHED_7_0_1, SMPL_SCHED_7_0_2, SMPL_SCHED_7_0_3. (13)</td>
</tr>
<tr>
<td>Event Rule</td>
<td>SMPL_FILTER_RULE (1)</td>
</tr>
</tbody>
</table>

You can display each object by running the composer command interface. For specific information on the syntax of the composer interface, see IBM Tivoli Workload Scheduler Version 8.4 Reference Guide.

Overview of the scheduling scenarios

The following table describes the topics covered in each scenario. Each scenario is a separate script file.

You can choose to run one or more scenarios in any order, but scenario one must always be the first one in the sequence.
Table 24. List of scheduling scenarios

<table>
<thead>
<tr>
<th>Script name/Scenario</th>
<th>Topics</th>
</tr>
</thead>
</table>
| Scenario1.0.bat (Windows) Scenario1.0.sh (UNIX): Scenario 1 | Creating the production plan and viewing plan contents  
**Note**: This scenario is a prerequisite for all other scenarios in your sequence. |
| Scenario2.0.bat (Windows) Scenario2.0.sh (UNIX): Scenario 2 | Administrative commands: starting and stopping Tivoli Workload Scheduler processes |
| Scenario3.0.bat (Windows) Scenario3.0.sh (UNIX): Scenario 3 | Scheduling basics: How jobs are scheduled, run order of jobs |
| Scenario4.0.bat (Windows) Scenario4.0.sh (UNIX): Scenario 4 | Advanced Scheduling: prompt, file, and resource dependencies |
| Scenario5.0.bat (Windows) Scenario5.0.sh (UNIX): Scenario 5 | Time dependencies and run cycles |
| Scenario6.0.bat (Windows) Scenario6.0.sh (UNIX): Scenario 6 | Job submission (jobs, job streams, ad-hoc jobs) |
| Scenario7.0.bat (Windows) Scenario7.0.sh (UNIX): Scenario 7 | Recovery options and recovery jobs |
| Scenario8.0.bat (Windows) Scenario8.0.sh (UNIX): Scenario 8 | Event driven scheduling |

Creating and working with the production plan

After you have successfully populated the database, you are ready to run the first scenario which creates the production plan. The production plan contains the database objects (jobs and job streams) that are eligible for scheduling.

While all other scenarios can be run in no particular order, this scenario is a prerequisite to all others.

Most commands in the scenarios are given in their short form. Where this is the case, the full name of the command is shown in parentheses in each scenario description.

**Scenario 1: Creating the production plan and viewing plan contents**

Scenario tasks and concepts:
- Create the production plan
- View the contents of the plan

Commands used in the scenario in their run sequence:
1. JnextPlan
2. conman sc (showcpus)
3. Planman showinfo
4. conman ss @#SMPL@ (showschedules)
Running the scheduling scenarios

After creating the plan, the following scenarios use the tutorial objects in the database by scheduling them in the plan. Each scenario explains different scheduling concepts. For each command used in the scenarios, the output is displayed on screen.

**Note:** You can run the scenarios in any order, as each scenario uses different objects. However, if you want to run the same scenario more than once in your sequence, you must reset the plan and run scenario 1 again before you rerun the individual scenario. Perform these steps:

1. Run the following command:
   ```bash
   resetplan -scratch
   ```
2. Run the `Scenario1.0.bat` (Windows) or `Scenario1.0.sh` script depending on your environment.

**Scenario 2: Starting and stopping Tivoli Workload Scheduler processes**

This scenario performs some basic administrative tasks. After each stop or start command, the status is displayed on screen.

Scenario tasks and concepts:
- Stopping and starting the Tivoli Workload Scheduler engine
- Stopping and starting the event processor
- Stopping and starting the monitoring agent
- Viewing process status

Commands used in the scenario in their run sequence:
1. "conman stop"
2. "conman status"
3. "conman start"
4. "conman status"
5. "conman stopevtproc" (stopeventprocessor)
6. "conman startevtproc" (starteventprocessor)
7. "conman sc" (showcpus)
8. "conman stopmon;wait"
9. "conman startmon"
10. "conman sc" (showcpus)

For a detailed description of Tivoli Workload Scheduler processes and related commands, see *IBM Tivoli Workload Scheduler Version 8.4 Reference Guide*.

**Scenario 3: Scheduling basics: How jobs are scheduled, run order of jobs**

This scenario performs basic scheduling tasks by showing how jobs are scheduled and how to influence the scheduling sequence.

Scenario tasks and concepts:
- Running a job and a job stream on a workstation
- Viewing job status
Running the scheduling scenarios

- Viewing and changing the workstation limit
- Understanding the concept and purpose of dependent job streams and run order (FOLLOW)
- Viewing dependency resolution during job runs

Commands used in the scenario in their run sequence:
1. "conman ss @SMPL_SCHED_3@" (showscheduled)
2. "composer disp sched=@SMPL_SCHED_3_0_2"
3. "conman lc 10:noask" (limit)
4. "conman sc" (showcpus)
5. "conman sj @SMPL_SCHED_3_0_@.SMPL_JOB_3_0_@" (showjobs)
6. "conman sj @SMPL_SCHED_3_0_@.SMPL_JOB_3_0_@" (showjobs)

Scenario 4: Advanced scheduling: prompt, file, and resource dependencies

This scenario performs advanced scheduling tasks by showing different types of dependencies in action.

Scenario tasks and concepts:
- Viewing and managing prompt dependencies
- Viewing and managing resource dependencies
- Viewing and managing file dependencies
- Understanding resource contention between jobs

Commands used in the scenario in their run sequence:
1. "composer disp sched= @#SMPL_SCHED_4@
2. "conman ss @#SMPL_SCHED_4@" (showscheduled)
3. "conman sp @#SMPLPRM4" (showprompts)
4. "conman reply SMLPRM4y" (reply)
5. "conman sp @#SMLPRM4" (showprompts)
6. "conman sj @#SMPL_SCHED_4_0_0@.SMPL_JOB_4_0_@" (showjobs)
7. "conman sj @#SMPL_SCHED_4_0_0@.SMPL_JOB_4_0_@" (showjobs)
8. "conman sj @#SMPL_SCHED_4_0_5@" (showjobs)

Scenario 5: Time dependencies and run cycles

This scenario performs advanced scheduling using time dependencies and run cycles.

Scenario tasks and concepts:
- Managing time limits such as AT® time and UNTIL time
- Releasing a time dependency
- Using run cycles to plan scheduling activities

Commands used in the scenario in their run sequence:
1. "conman sj @#SMPL_SCHED_5_0_0@.SMPL_JOB_5_0_@" (showjobs)
2. "conman ddj @#SMPL_SCHED_5_0_1.SMPL_JOB_5_0_1;at;noask" (deldep)
3. "conman sj @#SMPL_SCHED_5_0_1.SMPL_JOB_5_0_1" (showjobs)
4. "conman rj @#SMPL_SCHED_5_0_1.SMPL_JOB_5_0_2" (release)
Running the scheduling scenarios

5. "conman sj @@SMPL_SCHED_5_0_1.SMPL_JOB_5_0_2" (showjobs)
6. "conman ss @@SMPL_SCHED_5-@" (showschedules)

Scenario 6: Manual submission of jobs, job streams, and commands

This scenario uses the submit command to insert jobs, job streams, and ad-hoc jobs in the plan.

Scenario tasks and concepts:
- Submitting a job in the current production plan
- Submitting a job stream in the current production plan
- Submitting a command in the current production plan
- Displaying the job, job stream and command status in the plan

Commands used in the scenario in their run sequence:
1. "conman sbj @@SMPL_JOB_SBJ;alias=SMPL_SBJ_ALIAS" (submit)
2. "conman sj @@JOBS.SMPL_ALIAS" (showjobs)
3. "conman sbs @@SMPL_SCHED_SBS;alias=SMPL_SBS_ALIAS" (submit)
4. "conman sj @@SMPL_SBS_ALIAS" (showjobs)
5. "conman sbd "ver"; logon=^SMPLUSER^;alias=SMPL_SBD_ALIAS" (submit)
6. "conman sj @@JOBS.SMPL_SBD_ALIAS" (showjobs)

Note: The value of the logon attribute in step 5 is specified by using a parameter object. For more information about parameters see Tivoli Workload Scheduler Reference Guide.

Scenario 7: Recovery options and recovery jobs

This scenario shows some examples of recovery options and recovery jobs.

Scenario tasks and concepts:
- Defining and using the STOP, CONTINUE and RERUN recovery options
- Understanding the use of recovery jobs to solve scheduling malfunctions

Commands used in the scenario in their run sequence:
1. "conman reply SMPLPRM7;y" (reply)
2. "conman sp SMPLPRM7" (showprompts)
3. "conman sj @@SMPL_SCHED_7_0_1.@" (showjobs)
4. "conman sj @@SMPL_SCHED_7_0_2.@" (showjobs)
5. "conman sj @@SMPL_SCHED_7_0_3.@" (showjobs)

Scenario 8: Event-driven scheduling

This scenario shows some examples of event driven scheduling.

Scenario tasks and concepts:
- Creating a rule and associating an action to the rule
- Understanding the different rule types: Filter, Sequence, and Collection rules
- Processing an action associated to a rule

Commands used in the scenario in their run sequence:
Running the scheduling scenarios

1. "composer disp erule=SMPL_FILTER_RULE" (display)
2. "planman deploy -scratch"
3. "conman sj @#JOBS.SMPL_SBJ_ALIAS2"

Removing the tutorial objects from the database

You can choose to keep the database objects in your environment to use them as templates for new objects. If instead, you want to completely remove all tutorial objects from the database, perform the following steps:

1. Go to the <TWS_home>/TWStutorial directory, where TWS_home is the home directory of the user for which you have installed Tivoli Workload Scheduler.
2. Launch the tutorial utility installation script as follows:
   • In a Windows environment:
     SAMPLEDBSETUP.CMD -uninstall
   • In a UNIX environment:
     sampledbsetup.sh -uninstall
Removing tutorial objects
Chapter 5. Upgrading a version 8.2 or 8.2.1 master domain manager instance

This chapter describes how to upgrade master domain managers and backup master domain managers from Tivoli Workload Scheduler 8.2 or 8.2.1 to version 8.4. If you have components of Tivoli Workload Scheduler that are at an earlier version than 8.2 you must first upgrade them to version 8.2 or 8.2.1 before upgrading to version 8.4.

The upgrade from version 8.3 to version 8.4 is described in Chapter 6, “Upgrading a version 8.3 master domain manager instance,” on page 111.

To upgrade agents from all supported versions, refer to the procedures described in Chapter 7, “Upgrading agents,” on page 125.

The upgrade is described in the following topics:

- “Upgrading Overview” on page 77
- “Deciding how to upgrade the master domain manager” on page 78
- “Preparing the upgrade environment” on page 84
- “Performing a parallel upgrade” on page 84
- “Performing a direct upgrade” on page 94
- “Data import - problem resolving” on page 103
- “Post-upgrade issues” on page 110

Upgrading Overview

This section provides an overview of the upgrade of an existing instance of Tivoli Workload Scheduler version 8.2 or 8.2.1 (hereafter called 8.2.x).

The overview is provided under the following topics:

- “Upgrading from the top or from the bottom” on page 78
- “Component upgrade procedures” on page 78

Upgrading from the top or from the bottom

Tivoli Workload Scheduler 8.4 supports backward compatibility so you can upgrade your network in either of the following ways:

Top-down

Upgrade the master domain manager and backup master domain manager, and then progressively upgrade the agents. In this case, many of the new functions introduced in version 8.4 becomes available for each agent as it is upgraded. The disadvantage is that the same level of function is not available to all agents at the same time.

Bottom-up

Upgrade the agents first, and when all have been upgraded, upgrade the master domain manager and backup master domain manager. In this case, many of the new functions introduced in version 8.4 are not be available until the whole network has been upgraded.
Overview of upgrade

Component upgrade procedures

The following shows the components of your 8.2.x system, and where to find the upgrade procedure:

Master domain manager and backup master domain manager
To upgrade the master domain manager and backup master domain manager (if you use one), follow these main steps:

1. Choose the direct or parallel upgrade method
   Choose between a direct or a parallel upgrade. These alternatives are described in “Master and backup master domain manager upgrade scenarios” on page 79. The choice between parallel and direct upgrade is meaningful only for the master domain manager.

2. Decide how to upgrade your database
   Make the required decisions about the new relational database you will be using in version 8.4, and how to import the old data into it, following the guidelines in “Deciding how to upgrade the database” on page 80.

3. Perform the upgrade
   Follow the instructions in either “Performing a parallel upgrade” on page 84 or “Performing a direct upgrade” on page 94, as appropriate.

4. Complete the Security Configuration
   Follow the instructions in “Building the final security file for the new environment” on page 102, as appropriate.

5. Post-upgrade issues
   There are some additional configuration issues that must be resolved after the upgrade is complete. See “Post-upgrade issues” on page 110.

domain manager, fault-tolerant agent, standard agent,
To upgrade a domain manager or agent, follow the procedures in Chapter 7, “Upgrading agents,” on page 125.

Job Scheduling Console
You cannot upgrade any version of the Job Scheduling Console prior to version 8.3. Instead you must perform a fresh installation, as described in Tivoli Workload Scheduler: Job Scheduling Console User’s Guide.

Connector for the Job Scheduling Console
You cannot upgrade any version of the connector prior to version 8.3. Instead you must perform a fresh installation, as described in “Adding a new feature” on page 58.

Deciding how to upgrade the master domain manager

There are various factors that you must take into consideration before starting to upgrade your master domain manager. These factors are described in the following topics:

- “Master and backup master domain manager upgrade scenarios” on page 79
- “Deciding how to upgrade the database” on page 80
Master and backup master domain manager upgrade scenarios

Choose one of these two upgrade scenarios:

- **“Parallel upgrade - minimizing the impact on scheduling”**
- **“Direct upgrade - minimizing the work to be done and the resource usage”** on page 80

### Parallel upgrade - minimizing the impact on scheduling

<table>
<thead>
<tr>
<th>Steps</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Install a fresh master domain manager, creating a new database.</td>
<td>• Your old environment is operational throughout the entire upgrade process.</td>
<td>• It is a longer process that involves some manual configuration activities.</td>
</tr>
<tr>
<td>2. Import the scheduling and configuration data from the previous version.</td>
<td>• Each step is a single unit that can be reversed.</td>
<td></td>
</tr>
<tr>
<td>3. Complete your migrated environment by solving any data import problems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Complete your security file by merging old and new security settings</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Switch the domain manager to the new (fresh) master</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Deciding where to install the new master domain manager and the relational database (RDBMS)

The fresh installation of the master domain manager can be made on the same computer as the existing master domain manager or on a new computer. The new relational database can be installed on the same computer as the master domain manager (in this case a database client) or on a different one (database client or server). Consider the following points:

- If you choose a new computer for the master domain manager you provide a new and better-performing platform. However, to upgrade the old Tivoli Workload Scheduler database you will need to mount it as a file system on the new computer, or make a file transfer of the old database files to the new computer.
- Choosing the same computer for the master domain manager makes the old database upgrade easier.
- Installing the relational database on a separate computer from the master domain manager is good for the overall performance, provided that the link between the master domain manager and the relational database server computer is high speed.
- Installing the relational database on a separate computer from the master domain manager allows you to recover immediately from any failures. You can switch to your backup master domain manager without any loss of data.
- If you choose to install the new master domain manager instance on the same computer, install it with a different TWSUser from the existing instance.

The parallel upgrade is described in “Performing a parallel upgrade” on page 84.
Overview of upgrade

Direct upgrade - minimizing the work to be done and the resource usage

<table>
<thead>
<tr>
<th>How its done</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Unlink the old master domain manager</td>
<td>• Quicker and simpler than the parallel upgrade.</td>
<td>• Scheduling might be delayed for those activities involving the master domain manager</td>
</tr>
<tr>
<td>2. Upgrade the master domain manager, automatically importing the scheduling and configuration data from the previous version. <strong>Note:</strong> If you prefer, the data import can be performed as a separate step, giving you more control over it.</td>
<td>• Does not require the use of additional computer resources</td>
<td></td>
</tr>
<tr>
<td>3. Resolve any data import problems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Optionally export the Tivoli Management Framework users</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Complete your migrated environment by merging old and new security file settings.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. Rebuild the plan to make the upgraded master domain manager operational.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Optionally upgrade the backup master domain manager.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The direct upgrade is described in “Performing a direct upgrade” on page 94.

Deciding how to upgrade the database

The upgrade of the master domain manager from version 8.2.x not only upgrades the component, but also imports the version 8.2.x Mozart database into a new RDBMS, using DB2 or Oracle support. This is the most complex and time-consuming part of the upgrade.

The main advantage of this new infrastructure is that using an RDBMS avoids data inconsistencies. Furthermore, you can separate the database onto a different computer, helping to spread the workload and improving the performance of the master domain manager. It also enables you to designate a backup master domain manager that is configured to access the same database. Switching the master domain manager is now a much quicker process with reduced risk of data loss.

The import of the database can be performed automatically by the upgrade process or manually after the master domain manager is upgraded. The manual process is performed object type by object type, giving you maximum control.

Before you upgrade your installation, read the topics in this section. Topics are:

- “Choosing the Relational Data Base Management System to host the Tivoli Workload Scheduler database”
- “Understanding the steps in the data import” on page 81
- “Data import scenarios” on page 82

Choosing the Relational Data Base Management System to host the Tivoli Workload Scheduler database

The considerations that help you to choose the appropriate RDBMS (DB2 or Oracle) that will host the new Tivoli Workload Scheduler database are the same for an upgraded instance of Tivoli Workload Scheduler as they are for the installation
of a fresh instance. Therefore, read the information in “Supported Relational Data Base Management Systems” on page 12 and then return to this point and continue with the next section.

Understanding the steps in the data import
The data import requires the following steps. These steps are performed irrespective of whether you are importing the data manually, or the upgrade wizard is doing them for you. If you are using the update wizard for a direct upgrade, the following steps are performed automatically.

Import the configuration files
Your 8.2.x instance has two configuration files that must be migrated:

Run number file
This field contains the run number used by Symphony and is automatically incremented every time you run Jnextday. It is used in much the same way in version 8.4, but to preserve continuity you must import it.

Globalopts file
This contains the global options used by the master domain manager. In version 8.4 it is replaced by a database table, but the values must be imported.

Version 8.4 is supplied with the optman miggrunnb and optman miggopts subcommands that import these files.

Export the object data (not Windows users) from the Mozart database
The object data is extracted from the 8.2.x database using a specially enhanced version of the 8.2.x composer, called composer821, which is bundled with version 8.4. Do not use the original composer command for the migration. The create subcommand is used to create text files that contain the object data definitions.

Export the Windows user data from the Mozart database
If you use composer or composer821 to export the Windows users in the database, they are exported without passwords. This is the normal, secure, behavior of composer. For this reason, a migration utility, migrutility get_users, has been provided which extracts the Windows user data with their passwords (encrypted).

Export the Tivoli Management Framework user data from the Security file
In your version 8.2.x Security file, you might have Tivoli Management Framework administrators defined (used to assign rights to users who access the Job Scheduling Console). These must be migrated by a tool that substitutes each Tivoli Management Framework administrator with a local user account associated with those users. The users are exported first using the dumpsec command on the 8.2.x system. Secondly, the migrfwkuser utility does the account substitution. The output of the migrfwkuser utility is a security file containing the migrated users and your existing settings. You must then merge this information with the new version 8.4 security settings to build your final security file.

Import the object data into the new database
The object and Windows user data in the text files is imported into the new database using the datamigrate utility. This follows the stricter validation of the new database and identifies any logical discrepancies that were caused by the less strict validation in the 8.2.x system. In nearly all cases the utility is able to modify some aspect of the object and add it to the new database. A report of the discrepancies is produced so that you
Choosing your RDBMS

can fix the problems. You can choose to accept the way that the tool has dealt with the problem, or use the facilities of version 8.4 to modify the object.

For example, when mapping job definitions, if the workstation of the job does not exist, the job is not created in the database. However, if the recovery job does not exist, the job is created but a warning is issued.

The datamigrate utility has these options
• To migrate directly from the version 8.2.x database, or from the text files created by composer821 and migrutility get_users.
• To migrate each object type one-by-one, or to import all object data in one command.

Notes:
1. If an 8.2.x workstation has the same name of the new Tivoli Workload Scheduler 8.4 master domain manager, it will not be migrated. All jobs defined on that workstation will be defined under the master domain manager workstation
2. In the direct upgrade scenario, the workstation definition of the old master domain manager will be replaced by a new definition of a Tivoli Workload Scheduler master with the same workstation name but with default values. Before you begin the upgrade you should save the old master domain manager definition so that you can restore these values to your new master domain manager.
3. In a parallel upgrade scenario, if the new master domain manager workstation has the same name of the old master domain manager workstation, the tool will not migrate the old master domain manager definition.
4. The data import in a large environment or an environment with slow network speeds, may take some hours to complete, whether you do the import manually or automatically

Import the Tivoli Management Framework user data to the Security file
The makesec command is used to create the final Security file from the file created by the migrfwkuser utility merged with the new version 8.4 security settings.

Data import scenarios
Given the different options described in the previous sections, there are a number of data import scenarios, depending on whether you want a parallel or direct upgrade, and how closely you want to follow and control the activity:

Manually import data for a parallel upgrade
For a parallel upgrade, the manual import can be made in either of these modes:

Manual import direct from Mozart - scenario PD
If the new and the old instances of Tivoli Workload Scheduler are on the same computer, or you can mount the file system of the version 8.2.x instance of Tivoli Workload Scheduler on the system where the version 8.4 instance of Tivoli Workload Scheduler is installed, you can perform the manual import directly from the old Mozart database to the new RDBMS. To do this the two workstations must have the same system byte order.
1. Migrate the configuration files to the new database
Choosing your data import method

2. If you have Tivoli Management Framework users in version 8.2.x, export them to text files.
3. Directly import all the object data into the new database.
4. Complete your security configuration for the new environment.

This procedure is described in “Parallel 2D: Manually importing the Mozart data directly - scenario P2D” on page 85.

Manual import from an unlinked system - scenario PU

If the new and the old instances of Tivoli Workload Scheduler are on different computers and you cannot mount the file system of the previous instance on the new instance, you do the import in these stages:

1. Migrate the configuration files to the new database by copying the files from the instance of Tivoli Workload Scheduler 8.2.x to the version 8.4 instance, and then using the version 8.4 commands to import them into the 8.4 database.
2. Export the object data (not Windows users) from version 8.2.x into text files.
3. Export the version 8.2.x Windows user data into a text file.
4. If you have Tivoli Management Framework users in version 8.2.x, export them to text files.
5. Move all the text files to the system where the version 8.4 master domain manager is installed.
6. Import all the object data into the new database from the text files.
7. Complete your security configuration for the new environment.

This procedure is described in “Parallel 2U: Manually importing the 8.2.x data indirectly from an unlinked system - scenario P2U” on page 88.

Manually import data for a direct upgrade

If you have chosen the direct upgrade, the upgrade wizard automatically exports the 8.2.x object data to text files in a directory of your choice. You can opt to manually import these files rather than to let the wizard do the automatic import for you.

Note: When you select to import the data manually, the upgraded master has no master workstation defined in the database and only the composer command is available.

The procedure is as follows:
1. If you have Tivoli Management Framework users in version 8.2.x, export them to text files.
2. Directly import all the object data into the new database.
3. Import the Tivoli Management Framework users into the database from the text files.

This procedure is described in “Direct 3: Optionnally importing the 8.2.x data manually” on page 99.

Automatic import for a direct upgrade

If you do not want to follow the import process but want it to run in
Choosing your data import method

background, you can select an automatic import during the upgrade of the existing instance of the master domain manager. You can check the status of the import after completion.

Preparing the upgrade environment

This section describes the tasks you need to consider to prepare the 8.2.x environment for an upgrade.

Linux kernel
If you are upgrading in a Linux environment that uses the LD_ASSUMEKERNEL=2.4.1 environment variable, upgrade to Tivoli Workload Scheduler 8.4 in a shell that also uses the LD_ASSUMEKERNEL=2.4.1 environment variable.

Ensure your current Tivoli Workload Scheduler installations are in correct state
When you are upgrading your current environment, make sure the software package is in the COMMIT state (not in the UNDOABLE state). If it is in the UNDOABLE state, you must accept it to change its state to COMMIT, before you upgrade to version 8.4.

Performing a parallel upgrade

This section describes how to upgrade your environment using a parallel scenario. The parallel scenario consists of the following procedures:

- “Parallel 1: Installing a new version 8.4 master domain manager”
- “Parallel 2D: Manually importing the Mozart data directly - scenario P2D” on page 85
- “Parallel 2U: Manually importing the 8.2.x data indirectly from an unlinked system - scenario P2U” on page 88
- “Parallel 3: Data import- resolving problems” on page 93
- “Parallel 4: Switching the master domain manager” on page 93

Parallel 1: Installing a new version 8.4 master domain manager

The following sequence of steps describe the parallel upgrade scenario.

Parallel 1 a: Performing a fresh master domain manager installation
Install a master domain manager either on the same workstation where the existing master domain manager was installed, or on a different one. See “Deciding where to install the new master domain manager and the relational database (RDBMS)” on page 79 for information on the advantages and disadvantages of these options.

See Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41 for instructions on how to do this, bearing the following points in mind:

- If you choose to install the new instance on the same computer, install it with a different TWSUser from the existing instance.
- The workstation name of the new master domain manager must be different from the workstation name of the old master domain manager.
Parallel upgrade procedure

• If you want to have the database server support on Oracle you should install it before starting the installation of the master domain manager. During the installation you will identify the installed instance.

• Alternatively, if you want to have the database server support on DB2, and want it installed on a different computer from the master domain manager, you should install it first. During the installation you will be asked whether you want to install the DB2 server or client. Choose the "DB2 client" and then identify the installed DB2 server instance to connect to.

Parallel 1 b: Creating a workstation definition for new master domain manager in old domain

After the installation has completed, use composer in version 8.2.x to create a definition for the new master domain manager as a full status agent in the domain of the old master domain manager. The new workstation definition must be activated by a new plan creation before proceeding with the upgrade.

Parallel 2D: Manually importing the Mozart data directly - scenario P2D

If the new and the old instances of the master domain manager are on the same system, or you can mount the file system of the old instance on the new instance, follow these procedures:

• “P2D-1: Importing object data directly from Mozart”
• “P2D-2: Optionally exporting Tivoli Management Framework user data from the security file” on page 86
• “P2D-3: Completing the security configuration for the new environment” on page 87
• “P2D-4: Migrating the run number and global options” on page 87

P2D-1: Importing object data directly from Mozart

Use this procedure to import object data from a previous Tivoli Workload Scheduler version into the Tivoli Workload Scheduler 8.4 database in a parallel upgrade. When the Tivoli Workload Scheduler is on another workstation, you must mount the directory of the version 8.2.x environment on the version 8.4 system. To do this the two workstations must have the same system byte order. If they do not you must use the indirect import option (see “Parallel 2U: Manually importing the 8.2.x data indirectly from an unlinked system - scenario P2U” on page 88).

To import the data directly, follow these steps:

1. On the version 8.4 system, log in as a user that has full access to the database of both the old and the new Tivoli Workload Scheduler environments.

2. Set the Tivoli Workload Scheduler environment using the tws_env command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.

3. Use the datamigrate command to import the data directly from the existing Mozart file.

This step can be performed object type by object type, or in a single command for all object types:

Importing object data directly from Mozart in steps

The syntax of the command is as follows:

datamigrate object_type -path TWS_8.2.x_main_dir [-tmp_path temp_path]

where:
Parallel upgrade: direct manual data import

**object_type**

Is the type of object you are importing. Possible values are:
- calendars
- jobs
- job streams
- parms
- prompts
- resources
- topology
- users

You must run the command for all the object types indicated.

**TWS_8.2.x_main_dir**

Indicates the root directory of the previous Tivoli Workload Scheduler version.

**temp_path**

Is the temporary path where datamigrate stores the files during the migration process. The default is $TWS_home/tmp in UNIX systems and $TWS_home\tmp in Windows systems.

**P2D-2: Optionally exporting Tivoli Management Framework user data from the security file**

If you have customized user security settings based on Tivoli Management Framework Administrator ids rather than user ids in your security file, you can perform the following steps to transfer the current settings to a file which you can then import into your new Tivoli Workload Scheduler 8.4 environment.

To extract the Tivoli Management Framework users, perform the following steps:

1. Log in as the TWSuser.
2. Set the Tivoli Management Framework environment:
   - On UNIX, run `setup_env.sh` from `/etc/Tivoli`
   - On Windows, run `setup_env` from `c:\windows\system32\drivers\etc\Tivoli`
3. Set the Tivoli Workload Scheduler environment using the `tws_env` command. See the *Tivoli Workload Scheduler: Reference Guide* for full information about the command.
4. Run the Tivoli Workload Scheduler utility `dumpsec` to export the user information to a flat text file (`input_security_file`) as follows:
   ```
dumpsec > input_security_file
   ```
   where:
Parallel upgrade: direct manual data import

input_security_file

Is the text file created by the dumpsec command.

See the Tivoli Workload Scheduler: Reference Guide for full information about the command.

5. Locate the migrfwkuser utility command in the following tar file on the appropriate installation CD (see “The installation CDs” on page 21 for details):

CDn\operating_system\utilities\migrtool.tar

6. Uncompress the tar file in a directory on the version 8.2.x environment.

7. On Windows systems only run the bash command

8. Run set_env on Windows or ./set_env on UNIX

9. Run the migrfwkuser script as follows:

migrfwkuser -in input_security_file -out output_security_file
[-cpu workstation] [-hostname local_hostname]

where:

input_security_file

Is the file created using the dumpsec command in step 4.

output_security_file

Is the security file that is created by the migrfwkuser script.

workstation

Is the name of the local workstation where the login data added by the tool is defined. If you do not specify a workstation the data is taken from a localopts file if present in the same directory where the migrfwkuser script is located. If there is no localopts file, the workstation is set to the first 8 characters of the local host name.

local_hostname

Is the fully qualified host name of the Tivoli Management Framework users to be extracted. Login data for users with this host name or with this host name and domain name and the host name valid for all computers are extracted. If you do not specify the local host name, migrfwkuser retrieves the host name from the local computer and matches login data for computers with that host name and any domain name.

Note: After you run the command, the output_security_file contains only the framework users and the user definitions of your 8.2.x environment. You must manually merge this information with the new Tivoli Workload Scheduler 8.4 security settings before you import your final security file into the new 8.4 environment.

P2D-3: Completing the security configuration for the new environment

Before building your final security file, you must complete your security configuration. Follow the steps described in “Building the final security file for the new environment” on page 102.

P2D-4: Migrating the run number and global options

Use this procedure to migrate the Tivoli Workload Scheduler run number and globalopts files to version 8.4.

1. Log in as a user that has full access to the database of both the old and the new Tivoli Workload Scheduler environments.
Parallel upgrade: direct manual data import

2. Set the Tivoli Workload Scheduler environment using the `tws_env` command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.

3. Use the `optman` command to import the installation run number and global options.
   The syntax of the command is as follows:
   ```
   optman miggrunnb TWS_8.2.x_main_dir
   optman miggopts TWS_8.2.x_main_dir
   ```

   where:
   - `TWS_8.2.x_main_dir`
     Indicates the root directory of the previous Tivoli Workload Scheduler version.

   Proceed to “Parallel 3: Data import - resolving problems” on page 93

Parallel 2U: Manually importing the 8.2.x data indirectly from an unlinked system - scenario P2U

If the new and the old instances of the master domain manager are not on the same system, and you cannot mount the file system of the old instance on the new instance, follow these procedures:

- **P2U-1: Exporting the 8.2.x object data to flat text files**
- **P2U-2: Exporting the 8.2.x Windows user data to text files** on page 90
- **P2U-3: Moving the text files to the system where the new master domain manager is installed** on page 90
- **P2U-4: Importing object data from exported data files** on page 90
- **P2U-5: Optionally exporting Tivoli Management Framework user data from the security file** on page 91
- **P2U-6: Completing the security configuration for the new environment** on page 93
- **P2U-7: Migrating the run number and global options** on page 93

**P2U-1: Exporting the 8.2.x object data to flat text files**

When you are performing a parallel upgrade you must first manually export the old data to flat text files and then import the flat text files into the new database. This step describes how to export the data manually.

The data export is performed by a special version of the `composer` command, called `composer821`. The normal 8.2.x version of `composer` must not be used. The `composer821` command is located on the appropriate installation CD according to the operating system where the instance of Tivoli Workload Scheduler you are upgrading is installed. To export the data using the `composer821` command, perform the following steps:

1. Log in as the `TWSuser`.
2. Set the Tivoli Workload Scheduler environment using the `tws_env` command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.
3. Locate the file on the appropriate installation CD (see “The installation CDs” on page 21 for details):
   ```
   C:|\operating_system\bin\composer821
   ```
4. Copy the file into the directory where the old version 8.2.x composer is installed.

5. Assign to composer821 the same rights that the old composer has.

6. Use the composer821 create command to export the data. The data you export consists of the following:
   • Topology (workstations, workstation classes, and domains)
   • Prompts, calendars, parameters, resources
   • Jobs
   • Job streams

   Note: Users are not migrated with composer create. See “P2U-2: Exporting the 8.2.x Windows user data to text files” on page 90 for details.

The syntax of the command is as follows:

```
composer821 create topology_filename from cpu=@
composer821 create prompts_filename from prompt
composer821 create calendar_filename from calendar
composer821 create parms_filename from parms
composer821 create resources_filename from resources
composer821 create jobs_filename from jobs=@@
composer821 create scheds_filename from sched=@@
```

where:

- **topology_filename**
  - Is the name of the file that is to contain the topology data of the Tivoli Workload Scheduler instance you are upgrading (from cpu=@ indicates all workstations, workstation classes, and domains).

- **prompts_filename**
  - Is the name of the file that is to contain the prompts of the Tivoli Workload Scheduler instance you are upgrading (from prompt indicates all prompts).

- **calendar_filename**
  - Is the name of the file that is to contain the calendars of the Tivoli Workload Scheduler instance you are upgrading (from calendar indicates all calendars).

- **parms_filename**
  - Is the name of the file that is to contain the parameters of the Tivoli Workload Scheduler instance you are upgrading (from parms indicates all parameters).

- **resources_filename**
  - Is the name of the file that is to contain the resources of the Tivoli Workload Scheduler instance you are upgrading (from resources indicates all resources).

- **jobs_filename**
  - Is the name of the file that is to contain the jobs of the Tivoli Workload Scheduler instance you are upgrading (from jobs=@@ indicates all jobs).

- **scheds_filename**
  - Is the name of the file that is to contain the job streams of the Tivoli Workload Scheduler instance you are upgrading (from scheds=@@ indicates all schedules).

The output files are used in the import step.
Parallel upgrade: indirect manual data import

P2U-2: Exporting the 8.2.x Windows user data to text files

The composer821 create option for Windows users exports user details without their passwords. To include the Windows user passwords, follow these steps:

1. Log in as the TWSuser.
2. Set the Tivoli Workload Scheduler environment using the tws_env command. See the Tivoli Workload Scheduler: Reference Guide for detailed information about the command.
3. Clean up the Windows user definitions, eliminating users that are no longer valid. The Tivoli Workload Scheduler: Reference Guide describes how to remove user definitions from the database.
4. Locate the migrutility utility command in the following tar file on the appropriate installation CD (see "The installation CDs" on page 21 for details). 
   
   CD\operating_system\utilities\migrtool.tar
5. Uncompress the file and place it in a directory where you want to save the Windows users.
6. Use the command as follows:

```
migrutility get_users TWS_8.2.x_user_mozart_file users_filename
```

where:

- **TWS_8.2.x_user_mozart_file**
  - The complete path to file userdata located in <TWShome>/network/
    userdata.
- **users_filename**
  - A name of your choice for the output file to be created by migrutility. It includes the encrypted passwords.

The migrutility command extracts the Windows users (and their passwords) from the Tivoli Workload Scheduler network and stores them in users_filename. You will need users_filename to import the Windows users into the RDBMS of Tivoli Workload Scheduler version 8.4.

P2U-3: Moving the text files to the system where the new master domain manager is installed

Move all the text files created in "P2U-1: Exporting the 8.2.x object data to flat text files" on page 88 and "P2U-2: Exporting the 8.2.x Windows user data to text files" to any directory in the system where the new master domain manager is installed.

P2U-4: Importing object data from exported data files

To perform this step you must have completed "P2U-1: Exporting the 8.2.x object data to flat text files" on page 88, "P2U-2: Exporting the 8.2.x Windows user data to text files," and "P2U-3: Moving the text files to the system where the new master domain manager is installed."

Perform these steps:

1. On the version 8.4 system, log in as a user that has access to the exported object data files and the new Tivoli Workload Scheduler environment.
2. Set the Tivoli Workload Scheduler environment using the tws_env command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.
3. Use the datamigrate command to import the data from the dumped files.

The syntax of the commands to use is as follows:
datamigrate -topology topology_filename [-tmppath temp_path]
datamigrate -prompts prompts_filename [-tmppath temp_path]
datamigrate -calendars calendars_filename [-tmppath temp_path]
datamigrate -parms parms_filename [-tmppath temp_path]
datamigrate -resources resources_filename [-tmppath temp_path]
datamigrate -users users_filename [-tmppath temp_path]
datamigrate -jobs jobs_filename [-tmppath temp_path]
datamigrate -schedules scheds_filename [-tmppath temp_path]

where:

topology_filename
  Is the name of the topology file created by composer821 in the export process.

prompts_filename
  Is the name of the prompts file created by composer821 in the export process.

calendars_filename
  Is the name of the calendars file created by composer821 in the export process.

parms_filename
  Is the name of the parameters file created by composer821 in the export process.

resources_filename
  Is the name of the resources file created by composer821 in the export process.

users_filename
  Is the name of the Windows users file created with the migrutility utility in the export process.

jobs_filename
  Is the name of the jobs file created by composer821 in the export process.

scheds_filename
  Is the name of the job streams file created by composer821 in the export process.

temp_path
  Is the temporary path where datamigrate stores the files during the migrate process. The default is <TWS_home>/tmp in UNIX systems and <TWS_home>\tmp in Windows systems.

P2U-5: Optionally exporting Tivoli Management Framework user data from the security file

If you have customized user security settings based on Tivoli Management Framework Administrator ids rather than user IDs in your security file, you can perform the following steps to transfer the current settings to a file which you can then import into your new Tivoli Workload Scheduler 8.4 environment.

To extract the Tivoli Management Framework users, perform the following steps:
1. Log in as the TWSUser.
2. Set the Tivoli Management Framework environment:
   - On UNIX, run setup_env.sh from /etc/Tivoli
   - On Windows, run setup_env from c:\windows\system32\drivers\etc\Tivoli
Parallel upgrade: indirect manual data import

3. Set the Tivoli Workload Scheduler environment using the `tws_env` command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.

4. Run the Tivoli Workload Scheduler utility `dumpsec` to export the user information to a flat text file (`input_security_file`) as follows:
   ```
   dumpsec > input_security_file
   ```

   where:
   ```
   input_security_file
   ```
   Is the text file created by the `dumpsec` command.

   See the Tivoli Workload Scheduler: Reference Guide for full information about the command.

5. Locate the `migrfwkuser` utility `command` in the following `tar` file on the appropriate installation CD (see “The installation CDs” on page 21 for details):
   ```
   CDn\operating_system\utilities\migrtool.tar
   ```

6. Uncompress the tar file in a directory on the version 8.2.x environment.

7. On Windows systems only run the `bash` command

8. Run `set_env` on Windows or `.set_env` on UNIX

9. Run the `migrfwkuser` script as follows:
   ```
   migrfwkuser -in input_security_file -out output_security_file
   [-cpu workstation] [-hostname local_hostname]
   ```

   where:
   ```
   input_security_file
   ```
   Is the file created using the `dumpsec` command in step 4.

   ```
   output_security_file
   ```
   Is the security file that is created by the `migrfwkuser` script.

   ```
   workstation
   ```
   Is the name of the local workstation where the login data added by the tool is defined. If you do not specify a workstation the data is taken from the `localopts` file if present in the same directory where the `migrfwkuser` script is located. If there is no `localopts` file, the workstation is set to the first 8 characters of the local host name.

   ```
   local_hostname
   ```
   Is the fully qualified host name of the Tivoli Management Framework users to be extracted. Login data for users with this host name or with this host name and domain name and the host name valid for all computers are extracted. If you do not specify the local host name, `migrfwkuser` retrieves the host name from the local computer and matches login data for computers with that host name and any domain name.

   **Note:** After you run the command, the `output_security_file` contains only the framework users and the user definitions of your 8.2.x environment. You must manually merge this information with the new Tivoli Workload Scheduler 8.4 security settings before you import your final security file into the new 8.4 environment.
**Parallel upgrade: indirect manual data import**

**P2U-6: Completing the security configuration for the new environment**
Before building your final security file, you must complete your security configuration. Follow the steps described in “Building the final security file for the new environment” on page 102.

**P2U-7: Migrating the run number and global options**
Use this procedure to manually migrate the Tivoli Workload Scheduler run number and globalopts files to version 8.4:

1. On the new master domain manager create a directory named mozart in a path different from <TWSHome>.
2. Copy the files <TWSHome>/mozart/globalopts and <TWSHome>/mozart/runmsgno to the newly created mozart directory on the master domain manager.
3. Log in to the new master domain manager as a user that has full access to the database of both the old and the new Tivoli Workload Scheduler environments.
4. Set the Tivoli Workload Scheduler environment using the tws_env command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.
5. Use the optman command to import the installation run number and global options.
   The syntax of the command is as follows:
   
   ```
   optman migrunnb config_files_dir
   optman miggopts config_files_dir
   ```
   
   where:
   
   `config_files_dir`  
   Indicates the directory where you placed the version 8.2.x configuration files in step 1

**Parallel 3: Data import- resolving problems**
At this point in the parallel upgrade you should check the quality of the migrated data, as described in “Data import - problem resolving” on page 103. When this process is complete, return to this parallel upgrade procedure, the next step of which is “Parallel 4: Switching the master domain manager.”

**Parallel 4: Switching the master domain manager**
To switch from the old master domain manager to the new one, perform the following steps:

1. On the old master domain manager, run the following command:
   
   ```
   conman switchmgr MASTERDM;new_master_cpu_name
   ```
   
   where `new_master_cpu_name` is the name of the workstation where the new master domain manager resides.
2. On the new master domain manager, ensure that the carry forward option is set to ALL, by running the following command:
   
   ```
   optman chg cf=ALL
   ```
   
   See “Setting global options” on page 143.
3. On the new master domain manager, create a plan with 0 extension period that begins at the end of the current plan, by running the following command:
   
   ```
   JnextPlan -for 0000
   ```
Data mapping - problem resolution

4. On the new master domain manager, reset the carry forward option to the value you assigned before running Step 2 on page 93.

5. If you have a final schedule in your old environment and want to continue using it in the new version 8.4 environment, submit the following commands on the new master domain manager:
   a. composer add $final
   b. conman cs old_master_cpu_name#final

   where old_master_cpu_name is the name of the workstation where the old master domain manager resides.
   c. conman sbs final

6. Because you are adding a new final schedule, it is important that the old final schedule does not run. To avoid this, either delete the old final schedule from the database or set the priority to 0. To delete the old final schedule, run the following command:

   composer del sched=old_master_cpu_name#final

   where old_master_cpu_name is the name of the workstation where the old master domain manager resides.

Installing a fresh backup master domain manager

As a last step, you can optionally install a backup master domain manager on a workstation other than the new master domain manager. If you already have a backup master domain manager in your old environment, you can upgrade it using the direct upgrade procedure described in the following sections.

To install a new backup master domain manager see Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41 for instructions on how to do this.

Performing a direct upgrade

This section describes how to upgrade your environment directly. The upgrade consists of the following procedures:

- “Direct 1: Unlinking the 8.2.x master domain manager from its agents and stopping it” on page 95
- “Direct 2: Upgrading the master domain manager” on page 95
- “Direct 3: Optionally importing the 8.2.x data manually” on page 99
- “Direct 4: Data import - resolving problems” on page 100
- “Direct 5: Optionally exporting Tivoli Management Framework user data from the security file” on page 100
- “Direct 6: Completing the security configuration for the new environment” on page 101
- “Direct 7: Rebuilding the plan for the upgraded master domain manager” on page 101
- “Direct 8: Upgrading the backup master domain manager” on page 102

Direct 1: Unlinking the 8.2.x master domain manager from its agents and stopping it

Before commencing the upgrade you must unlink all workstations from the master domain manager and stop it.

Follow these steps:
Performing a direct upgrade

1. Log in as the TWSuser.
2. Unlink all workstations in the domain:
   - From the Job Scheduling Console
     a. Run a Status of all Domains plan list. The Status of all Domains table opens.
     b. Right-click the domain you want to unlink and select Unlink Workstations from the pop-up menu.
   - From the command line of the master domain manager
     Issue the following command:
     conman "unlink @;noask"
3. Stop the master domain manager:
   - From the Job Scheduling Console
     a. Select the master domain manager workstation
     b. Right-click the workstation and select Stop from the pop-up menu.
   - From the command line of the master domain manager
     Issue the following command:
     conman "stop;wait"
4. From the command line of the master domain manager, stop the netman process as follows:
   - On UNIX, run:
     conman "shut"
     Note: Do not use the UNIX kill command to stop Tivoli Workload Scheduler processes.
   - On Windows, run the shutdown.cmd command from the Tivoli Workload Scheduler home directory.
5. Verify that all services and processes are not running, as follows:
   - On UNIX, type the command:
     ps -u TWSuser
     Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, JOBMAN, stageman.
   - On Windows, type the command:
     <drive>\unsupported\listproc.exe
     Where:
     <drive>
     is the TWS_home directory
     Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, stageman, JOBMON, batchup.
     Also, ensure that no system programs are accessing the directory or subdirectories, including the command prompt, and that in Windows Explorer the Administrative Tools\Services panel is not opened.

Direct 2: Upgrading the master domain manager
This section describes how to upgrade a master domain manager. When the upgrade procedure is successful it is not possible to roll back to the previous version. Rollback is only possible for upgrades that fail.
Performing a direct upgrade

You can upgrade a master domain manager using the following installation methods:
- "D2-W: Upgrading a master domain manager using the installation wizard"
- "D2-S: Upgrading the master domain manager using a silent installation" on page 98

D2-W: Upgrading a master domain manager using the installation wizard

During the upgrade procedure, the installation wizard backs up all the master data and configuration information, installs the new product code, and optionally dumps old scheduling data and configuration information. When the upgrade procedure is successful it is not possible to roll back to the previous version. Rollback is only possible for upgrades that fail.

This procedure can be used to upgrade either a master domain manager or a backup master domain manager. Steps or considerations which apply only for the master domain manager upgrade are indicated by "Master only:"

The steps are as follows:
1. If you are upgrading in a Windows environment, check that the Tivoli Token Server service is running.
2. Insert the installation CD according to the operating system. See "The installation CDs" on page 21.
3. Run setup for the operating system on which you are upgrading:
   - On Windows, WINDOWS\SETUP.exe
   - On UNIX and Linux, SETUP.sh or operating_system/SETUP.bin depending on whether you want to copy the images locally.
   - Alternatively, start the launchpad as follows and select the Tivoli Workload Scheduler installation:

      On Windows systems
      From the root directory of the CD run the launchpad.exe file.

      On UNIX systems
      From the root directory of the CD run the launchpad.sh file.
4. The installation wizard is launched. Select the installation wizard language. Click OK.
5. Read the welcome information and click Next.
6. Read and accept the license agreement. Click Next.
7. Select a previous instance of the product and component from the drop-down list and click Upgrade an instance of Tivoli Workload Scheduler. The components are recognized by an acronym in parentheses after the product name:
   - MDM  Master domain manager
   - BKM  Backup master domain manager

   Leave selected the check box Upgrade an instance of Tivoli Workload Scheduler. Click Next.
8. Master only: If you are installing a master domain manager, specify the information necessary to perform the upgrade. Provide the following data:
Upgrading a master domain manager using the installation wizard

Backup the previous Tivoli Workload Scheduler instance
Select whether to back up the previous instance.

Backup Destination Directory
When you select to back up the previous instance, specify the directory where the backup is located.

Export Destination Directory
The upgrade always exports the object data into text files, whether you want to import it automatically or manually. Specify the directory where you want the object data files to be saved. The default path is:
- $TMP/tws84/datamigration in UNIX systems
- %TEMP%\tws84\datamigration in Windows systems

Automatically import data from the version being upgraded
Clear this option only if you want to manually import the version 8.2.x object data into the RDBMS. For more information on making this choice see “Deciding how to upgrade the database” on page 80.

Click Next.

9. Type the password of the Tivoli Workload Scheduler user for which you want to upgrade the instance. Click Next.

10. Specify the following application server port information. You should accept the default values unless you know that the default ports are already in use. If you need to provide different values, any valid port number can be used:

HTTP Transport
The port for the HTTP transport. The default value is 31115.

HTTPS Transport
The port for the secure HTTP transport. The default value is 31116.

Bootstrap / RMI
The port for the bootstrap or RMI. The default value is 31117.

SOAP Connector
The port for the application server protocol SOAP connector. The default value is 31118.

SAS Port
The default value is 31119.

Server Authentication Port
The default value is 31120.

Server Mutual Authentication Port
The default value is 31121.

ORB Port
The default value is 31122.

Administration Port
The default value is 31123.

Administration Secure Port
The default value is 31124.

Event Processor Monitoring (EIF) Port
The default value is 31131. This port is not used by backup master domain managers. Click Next.
Upgrading a master domain manager using the installation wizard

Click Next. The wizard has to check that these ports are free, which might take a few seconds.

11. Specify the relevant RDBMS data in accordance with the database installation type or configuration you are performing. See “Installing a new master domain manager or backup master” on page 43 for the relevant information. You have to choose which type of database support (DB2 or Oracle), optionally choose a DB2 component to install, and supply the appropriate configuration information, which requires several panels.

12. Review the summary data. If there is a problem, click Back and check the data you have input.

If the data seems correct, before clicking Next to start the installation process, note these considerations:

• The installation is performed as a series of steps. If a step fails, you might be able to correct the problem and resume the installation. This means that if the installation fails, before taking any action, consult the Administration and Troubleshooting manual to understand the implications of any action.

• Every time you run an installation, you overwrite any recovery data that was stored in respect of any previous installation. This means that if a previous installation is incomplete, for any reason, the action of running a new installation, even if it is for a different component, will make it impossible for you to complete the previous installation (by resuming a failed step, for example).

• Master only: If you are installing a master domain manager and you have chosen the automatic data import, the import is performed as a separate step in the installation process. In a large environment or an environment with slow network speeds, this step might take some hours to complete. During this step, note that the wizard’s progress bar appears to be stopped at 97% complete while the import is being performed. This is normal. If you think the data import process might have failed, use system tools to determine if the datamigrate and composer processes are still running.

13. When the installation completes successfully, an appropriate panel is displayed. In case of an unsuccessful installation, see the Administration and Troubleshooting.

Master only: If you are installing a master domain manager and you chose to import data automatically, a panel might be displayed indicating that the data has not been migrated cleanly. In this case, check the information in the directory indicated in the panel. See “Data import - problem resolving” on page 103.

14. Click Finish.

D2-S: Upgrading the master domain manager using a silent installation

To upgrade a master domain manager using a silent installation follow the procedure described in “Performing a silent installation” on page 60 using the response file TWS84_UPGRADE_MDM.txt.

If you choose to import data automatically, and errors occur during the data migration process, Tivoli Workload Scheduler creates the following file:

UNIX     $TMP/tws84/migrationErrors
Windows  %TEMP%	ws84\migrationErrors

Check the information in the directory indicated in the file. See “Data import - problem resolving” on page 103.
For more information on choosing to import data automatically or manually, see “Deciding how to upgrade the database” on page 80

**Direct 3: Optionally importing the 8.2.x data manually**

If you did not import the data automatically during the upgrade, perform these steps:

1. On the upgraded master domain manager, set the Tivoli Workload Scheduler environment using the `tws_env` command. See the *Tivoli Workload Scheduler: Reference Guide* for detailed information about the command.

2. Define the master domain manager workstation using the `composer` command interface as follows:
   - On UNIX type the command:
     ```plaintext
     composer rep cpudef_unix
     ```
   - On Windows type the command:
     ```plaintext
     composer rep cpudef_wnt
     ```

   where:
   - `cpudef_unix` is the text file containing the workstation definition for the master domain manager in UNIX environments.
   - `cpudef_wnt` is the text file containing the workstation definition for the master domain manager in Windows environments.

   These files were created during the upgrade process.

3. Use the `datamigrate` command to import the data from the exported text files. These files were created by the installation wizard and stored in the path specified in the Export Destination Directory field. See “D2-W: Upgrading a master domain manager using the installation wizard” on page 96.

   The syntax of the commands to use is as follows:
   ```plaintext
   datamigrate -topology topology_filename [-tmppath temp_path]
datamigrate -prompts prompts_filename [-tmppath temp_path]
datamigrate -calendars calendars_filename [-tmppath temp_path]
datamigrate -parms parms_filename [-tmppath temp_path]
datamigrate -resources resources_filename [-tmppath temp_path]
datamigrate -users users_filename [-tmppath temp_path]
datamigrate -jobs jobs_filename [-tmppath temp_path]
datamigrate -schilds scheds_filename [-tmppath temp_path]
   ```

   where:
   - `topology_filename` is the name of the topology file created by `composer821` in the export process.
   - `prompts_filename` is the name of the prompts file created by `composer821` in the export process.
   - `calendars_filename` is the name of the calendars file created by `composer821` in the export process.
   - `parms_filename` is the name of the parameters file created by `composer821` in the export process.
Direct 4: Data import - resolving problems

At this point in the direct upgrade, check the quality of the imported data, as described in “Data import - problem resolving” on page 103. When this process is complete, return to this parallel upgrade procedure, the next step of which is "Direct 7: Rebuilding the plan for the upgraded master domain manager" on page 101.

Direct 5: Optionally exporting Tivoli Management Framework user data from the security file

If you have customized user security settings based on Tivoli Management Framework Administrator ids rather than user IDs in your security file, you can perform the following steps to transfer the current settings to a file which you can then import into your new Tivoli Workload Scheduler 8.4 environment.

To extract the Tivoli Management Framework users, perform the following steps:

1. Log in as the TWSUser.
2. Set the Tivoli Management Framework environment:
   - On UNIX, run setup_env.sh from /etc/Tivoli
   - On Windows, run setup_env from c:\windows\system32\drivers\etc\Tivoli
3. Set the Tivoli Workload Scheduler environment using the tws_env command. See the Tivoli Workload Scheduler: Reference Guide for full information about the command.
4. Run the Tivoli Workload Scheduler utility dumpsec to export the user information to a flat text file (input_security_file) as follows:

   dumpsec > input_security_file

   where:

   input_security_file
   Is the text file created by the dumpsec command.

   See the Tivoli Workload Scheduler: Reference Guide for full information about the command.
5. Locate the **migrfwkuser** utility command in the following tar file on the appropriate installation CD (see “The installation CDs” on page 21 for details):

   `C:\operating_system\utilities\migrtool.tar`

6. Uncompress the tar file in a directory on the version 8.2.x environment.

7. On Windows systems only run the **bash** command.

8. Run **set_env** on Windows or .**/set_env** on UNIX.

9. Run the **migrfwkuser** script as follows:

   ```
   migrfwkuser -in input_security_file -out output_security_file
   [-cpu workstation] [-hostname local_hostname]
   ```

   where:

   **input_security_file**
   Is the file created using the **dumpsec** command in step 4.

   **output_security_file**
   Is the security file that is created by the **migrfwkuser** script.

   **workstation**
   Is the name of the local workstation where the login data added by the tool is defined. If you do not specify a workstation the data is taken from a **localopts** file if present in the same directory where the **migrfwkuser** script is located. If there is no **localopts** file, the workstation is set to the first 8 characters of the local host name.

   **local_hostname**
   Is the fully qualified host name of the Tivoli Management Framework users to be extracted. Login data for users with this host name or with this host name and domain name and the host name valid for all computers are extracted. If you do not specify the local host name, **migrfwkuser** retrieves the host name from the local computer and matches login data for computers with that host name and any domain name.

   **Note:** After you run the command, the **output_security_file** contains only the framework users and the user definitions of your 8.2.x environment. You must manually merge this information with the new Tivoli Workload Scheduler 8.4 security settings before you import your final security file into the new 8.4 environment.

### Direct 6: Completing the security configuration for the new environment

Before building your final security file, you must complete your security configuration. Follow the steps described in “Building the final security file for the new environment” on page 102.

### Direct 7: Rebuilding the plan for the upgraded master domain manager

To configure the upgraded master domain manager, perform the following steps:

1. Ensure that the carry forward option is set to all, by running the following command:

   ```
   optman chg cf=ALL
   ```

   See “Setting global options” on page 143.
2. Create a plan with 0 extension period that begins at the end of the current plan, by running the following command:
   
   JnextPlan -for 0000

3. Reset the carry forward option to the value you assigned before running Step 1 on page 101.

4. If you have a final schedule in your old environment and want to continue using it in the upgraded environment, submit the following commands on the new master domain manager:
   a. composer rep $final 
   b. conman cs final
   c. conman sbs final

**Direct 8: Upgrading the backup master domain manager**

The upgrade procedures you use are the same procedures used to upgrade the master domain manager. Some fields do not apply to the backup master domain manager upgrade. These fields are clearly marked in the procedure.

The backup master domain manager can be upgraded using the installation wizard or using a silent installation. To upgrade using the installation wizard, see “D2-W: Upgrading a master domain manager using the installation wizard” on page 96.

To upgrade a backup master domain manager using a silent installation follow the procedure described in “Performing a silent installation” on page 60 using the response file TWS84_UPGRADE_BACKUP_MDM.txt.

**Building the final security file for the new environment**

The upgrade scenarios do not affect your current version 8.2.x security file which is left unchanged in your old installation. Version 8.4 introduces new security statements for the event management and reporting features. If you have specific security settings in your 8.2.x environment, these settings must be manually merged with the new settings before you build the final security file to be used in your new 8.4 environment. The statements you may have to add manually vary depending on your specific security settings and on whether you have chosen the parallel upgrade or the direct upgrade scenario.

**Parallel upgrade scenario**

The security file of your new master domain manager contains all the necessary entries for the new event management and reporting features. If you do not have any specific security settings or framework users in your old environment, you are ready to work directly with the new security file.

If you have run the `migrfwkuser` utility on your old security file, you need to merge the information contained in the `output_security_file` with the new security settings into a single text file and then import it using the `makesec` command. Perform the following steps:

1. On the new version 8.4 master domain manager log in as the `TWS_user` and set the Tivoli Workload Scheduler environment. Extract the new security file on the version 8.4 master using the following command:
   
   dumpsec > v84_sec_file

   where `v84_sec_file` is the text file created by the `dumpsec` command.
2. Transfer the output_security_file you obtained running the procedure described in “T2D-2: Optionally exporting Tivoli Management Framework user data from the security file” on page 86 from your old system to the new master domain manager system.

3. Edit the output_security_file and copy and paste the following statements from the v84_sec_file:

   REPORT NAME=@ ACCESS=DISPLAY
   EVENTRULE NAME=@ ACCESS=ADD,DELETE,D display,MODIFY,LIST,UNLOCK
   ACTION PROVIDER=@ ACCESS=DISPLAY,SUBMIT,USE,LIST
   EVENT PROVIDER=@ ACCESS=USE

   and the new Tivoli Workload Scheduler user from the initial statement, as shown in the following example:

   USER MAESTRO
   CPU=@+LOGON=TWS_user,root

   where TWS_user is the user of your Tivoli Workload Scheduler version 8.4 master domain manager installation. root or Administrator are the default users depending on whether you are in a UNIX or Windows environment. See the Tivoli Workload Scheduler: Reference Guide for further information about the security file definitions.

4. Save your changes to the output_security_file.

5. Build your final security file for your version 8.4 master domain manager using the makesec command:

   makesec output_security_file

Direct upgrade scenario

In the direct upgrade scenario, you have the version 8.2.x security file that must be merged with the security settings that support event management and reporting features. If you exported the Framework users and ran the migfwkuser utility, do the following:

1. Log in as TWS_user on your upgraded master domain manager and set the Tivoli Workload Scheduler environment. Edit the output_security_file obtained in the export step, and insert the following statements:

   REPORT NAME=@ ACCESS=DISPLAY
   EVENTRULE NAME=@ ACCESS=ADD,DELETE,D display,MODIFY,LIST,UNLOCK
   ACTION PROVIDER=@ ACCESS=DISPLAY,SUBMIT,USE,LIST
   EVENT PROVIDER=@ ACCESS=USE

2. If you do not have any framework users to migrate, use the dumpsec command to export your security settings to a flat file and insert the statements listed in the preceding step. Save the changes and build your final security file:

   makesec output_security_file

   where the output_security_file is the file you edited in step 1 or 2.

Data import - problem resolving

This section describes how you validate the data import and correct any inconsistencies.

The information in this section assumes that the data import has successfully completed. For problems relating to the running of the data import, refer to Tivoli Workload Scheduler Administration and Troubleshooting.
The topics in this section are as follows:

- "Data import log files"
- "Object data inconsistencies and recovery"

**Data import log files**

During the automatic data import the following log files are created in the `<TWS_home>/tmp` directory in UNIX environments or `<TWS_home>\tmp` in Windows environments. If you have chosen a manual import and have specified a different value in the `temp_path` parameter, the files are created in the path you indicated.

- `datamigrate_object_nnnnnn.log`
  Contains all the messages created during the import process, where `object` is the database object and `nnnnnn` is an identifier of each `datamigrate` run.

- `datamigrate_object_nnnnnn.err`
  Contains the error messages created during the data import process, where `object` is the database object and `nnnnnn` is an identifier of each `datamigrate` run.

The database `object` files are the following:

- CALENDARS
- JOBS
- PARMS
- PROMPTS
- RESOURCES
- SCHEDS
- TOPOLOGY
- USERS

You only need to fix situations that are reported as errors during the data import process.

**Note:** Because each error is logged for each step of the migration process the reported number of errors is greater than the true level.

**Object data inconsistencies and recovery**

This section describes how any inconsistencies in the imported 8.2.x data are mapped to 8.4, and any manual recovery actions required. These actions must be performed before you can commence scheduling with version 8.4.

In most cases the discrepancies between the two versions can be traced to inconsistencies in the previous database. The information in the `Mapping` column of each object-type table below tells you how the data import has dealt with any inconsistencies. You have two options for correcting these inconsistencies:

- Correct the inconsistency in Tivoli Workload Scheduler, version 8.4
- Correct the inconsistency in Tivoli Workload Scheduler, version 8.2.x and redo the import. If you want to take this option you must first remove the imported data from the database before redoing the import.

The mapping is organized by object type:

- "Mapping domains" on page 105
Inconsistencies in the imported data

- “Mapping workstations”
- “Mapping workstation classes” on page 106
- “Mapping job definitions” on page 106
- “Mapping Windows users” on page 107
- “Mapping resources” on page 107
- “Mapping job streams” on page 107
- “Mapping jobs” on page 108
- “Mapping dependencies” on page 108

Mapping domains

Table 25 lists the mapping and behavioral differences between 8.2.x and 8.4 domains and when a recovery action might be required.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domains with no manager</td>
<td>Definition allowed, domains not added to plan. Workstations are moved to the Master Domain; sub-domains are moved under the Master Domain.</td>
<td>As for 8.2.x</td>
<td>No mapping is required.</td>
</tr>
<tr>
<td>Domain has a manager defined in a different domain, or the manager is manager of more than one domain.</td>
<td>Possible (adding or changing workstation later). Compiler assigned to the latest domain definition.</td>
<td>Not possible.</td>
<td>If the domain refers to a workstation that exists, data import modifies the workstation to point to the domain. If the domain refers to a workstation that does not exist, the domain is left without a manager.</td>
</tr>
<tr>
<td>A parent domain does not exist.</td>
<td>Possible (removing the domain later). Compiler uses the Master Domain as parent.</td>
<td>Not possible.</td>
<td>Data import issues a warning and uses the Master Domain as parent.</td>
</tr>
</tbody>
</table>

Mapping workstations

Table 26 lists the mapping and behavioral differences between 8.2.x and 8.4 workstations and when a recovery action might be required.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workstation domain does not exist.</td>
<td>Possible (removing domain later). Compiler links the workstation to the Master Domain.</td>
<td>Not allowed.</td>
<td>Data import links the workstation to the Master Domain and issues a warning.</td>
</tr>
<tr>
<td>Workstation is a domain manager but is not a fault-tolerant agent or does not have full status set.</td>
<td>Possible as the compiler changes the workstation to a full status fault-tolerant agent.</td>
<td>The full status flag is automatically set to true when a Domain Manager is saved.</td>
<td>If the Domain Manager is a standard agent, data import issues a warning and the full status flag is set to true. If the Domain Manager is an extended agent the workstation is not promoted to an fault-tolerant agent.</td>
</tr>
</tbody>
</table>
Inconsistencies in the imported data

Table 26. Data and behavior mapping for workstations (continued)

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard agent with no domain but host specified.</td>
<td>Possible as the compiler sets the domain to the domain of the hosting workstation and moves the workstation under the Domain Manager of the domain.</td>
<td>The domain is mandatory and the host ignored.</td>
<td>If the domain is null and host is not, data import sets the domain as the hosting workstation domain.</td>
</tr>
<tr>
<td>Extended agent with domain specified.</td>
<td>Possible.</td>
<td>Domain is not allowed.</td>
<td>Data import ignores the domain.</td>
</tr>
<tr>
<td>Host of an extended agent or standard agent does not exist.</td>
<td>Possible (removing host workstation later). Extended agents are hosted by the MASTER workstation.</td>
<td>Not allowed.</td>
<td>Data import assigns the workstation to the MASTER workstation.</td>
</tr>
<tr>
<td>The workstation operating system is one of the following: • MPEV • MPEXL • MPIX • MPEIX • POSIX • HPUX • AIX</td>
<td>composer treats all of these operating systems as UNIX. Operating systems are modified to UNIX.</td>
<td>Data import considers all of the operating systems as UNIX.</td>
<td></td>
</tr>
</tbody>
</table>

Mapping workstation classes

Table 27 lists the mapping and behavioral differences between 8.2.x and 8.4 workstation classes and when a recovery action might be required.

Table 27. Data and behavior mapping for workstation classes

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Some or all of the workstations in the workstation class do not exist.</td>
<td>Possible (removing workstations later).</td>
<td>Not allowed.</td>
<td>If one or more of the workstations do not exist, data import ignores them, saves the workstation class and issues a warning. If no workstation exist the workstation class is created empty.</td>
</tr>
</tbody>
</table>

Note: If you have a version 8.2.x workstation class with a name longer than 128 bytes, the composer command treats the list member as two separate items and creates two incorrect workstation classes. If you used the composer821 command this problem should not occur.

Mapping job definitions

Table 28 on page 107 lists the mapping and behavioral differences between 8.2.x and 8.4 job definitions and when a recovery action might be required.
### Inconsistencies in the imported data

**Table 28. Data and behavior mapping for job definitions**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The workstation or workstation class does not exist.</td>
<td>Possible (removing workstation later). The compiler does not add the job to Symphony. Ad hoc subs goes into error. If the job is part of a job stream the job is added to the plan and remains in READY state. Any jobs depending on it, remain in the HOLD state.</td>
<td>Not allowed.</td>
<td>Data import does not add the job definition and issues an error. Data import proceeds.</td>
</tr>
<tr>
<td>The recovery job does not exist.</td>
<td>Possible (removing job definition later). The compiler does not add the job to the Symphony. Ad hoc subs goes into error.</td>
<td>Not allowed.</td>
<td>Data import ignores the recovery job and issues a warning.</td>
</tr>
</tbody>
</table>

**Mapping Windows users**

Table 29 lists the mapping and behavioral differences between 8.2.x and 8.4 Windows users and when a recovery action might be required.

**Table 29. Data and behavior mapping for Windows users**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Windows user workstation does not exist.</td>
<td>Possible (removing workstation later). The compiler adds the user, but it can not be used by any jobman.</td>
<td>Not allowed.</td>
<td>Data import ignores the Windows user and issues an error. Data import proceeds.</td>
</tr>
</tbody>
</table>

**Mapping resources**

Table 30 lists the mapping and behavioral differences between 8.2.x and 8.4 resources and when a recovery action might be required.

**Table 30. Data and behavior mapping for resources**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The workstation or workstation class of the resource does not exist.</td>
<td>Possible (removing workstation later). The resource are added to the plan.</td>
<td>Not allowed.</td>
<td>Data import ignores the resources and issues an error. Data import proceeds.</td>
</tr>
</tbody>
</table>

**Mapping job streams**

Table 31 lists the mapping and behavioral differences between 8.2.x and 8.4 job streams and when a recovery action might be required.

**Table 31. Data and behavior mapping for job streams**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The workstation or workstation class of the job stream does not exist.</td>
<td>Possible (removing workstation later). The compiler does not add the job stream.</td>
<td>Not possible.</td>
<td>Data import ignores the job stream and issues an error. Data import proceeds.</td>
</tr>
</tbody>
</table>
Inconsistencies in the imported data

Table 31: Data and behavior mapping for job streams (continued)

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The job stream refers to a free days calendar that does not exist.</td>
<td>Possible (removing the calendar later). The compiler adds the job stream to the plan.</td>
<td>Not possible.</td>
<td>Data import sets the free days calendar to null an issues a warning.</td>
</tr>
<tr>
<td>The calendar run cycle refers to a calendar that does not exist.</td>
<td>Possible (removing the calendar later). The compiler does not add the job stream to the plan.</td>
<td>Not possible.</td>
<td>Data import adds the job stream to the plan without that run cycle and issues a warning.</td>
</tr>
<tr>
<td>The job stream contains job definitions that are defined in different ways in multiple job streams.</td>
<td>Possible (adding new job streams containing the same job definition with different specifications). The latest job definition is added to the plan.</td>
<td>Not possible.</td>
<td>During the data import process the job definitions are discarded. No warning is issued because the job definitions are loaded before loading the job stream.</td>
</tr>
<tr>
<td>The job stream has a dependency on a file with a name that is longer than 28 characters.</td>
<td>Possible (the dependency is accepted by composer, but is deleted when Jnextday is run).</td>
<td>Not possible.</td>
<td>The job stream is imported, the file dependency is dropped and the priority is set to 0. Data import proceeds.</td>
</tr>
<tr>
<td>The job stream is defined on a workstation class and has one or more jobs also defined on a workstation class.</td>
<td>This scenario is improbable in a production environment. The compiler exits without completing plan creation.</td>
<td>Not possible.</td>
<td>Data import ignores the job stream and issues an error. Data import proceeds.</td>
</tr>
</tbody>
</table>

Mapping jobs

Table 32 lists the mapping and behavioral differences between 8.2.x and 8.4 jobs and when a recovery action might be required.

Table 32: Data and behavior mapping for jobs

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>The job has a dependency on a file with a name that is longer than 28 characters.</td>
<td>Possible (the dependency is accepted by composer, but is deleted when Jnextday is run).</td>
<td>Not possible.</td>
<td>The job stream is imported, the file dependency is dropped and the priority is set to 0. Data import proceeds.</td>
</tr>
</tbody>
</table>
| The job refers to a job definition that does not exist. | Possible (removing the job definition later). The compiler substitutes the job with a dummy job in the FAIL state. | Not possible. | Data import replaces the job definition with a dummy job definition and sets the job priority to 0, guaranteeing that successors are not run. The job name is: 
TWS_INITIAL_DUMMY_JOBDEFINITION

With the following definition:
TWS_INITIAL_DUMMY_JOBDEFINITION
"DCOMMAND DUMMY_TASK"
"STREAMLOGON DUMMY_USER"

Mapping dependencies

Table 33 on page 109 lists the mapping and behavioral differences between 8.2.x and 8.4 dependencies and when a recovery action might be required.
Table 33. Data and behavior mapping for dependencies

<table>
<thead>
<tr>
<th>Configuration</th>
<th>8.2.x behavior</th>
<th>8.4 behavior</th>
<th>Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>An external dependency refers to a job or a job stream that does not exist.</td>
<td>Possible, dependency is not added to the plan.</td>
<td>Not possible.</td>
<td>Data import removes the dependency and issues a warning.</td>
</tr>
<tr>
<td>An Inter network dependency uses a nonexisting workstation.</td>
<td>Possible, dependency is not added to the plan.</td>
<td>Not possible.</td>
<td>Data import removes the dependency and issues a warning.</td>
</tr>
<tr>
<td>A dependency refers to a prompt in a job or a job stream that does not exist.</td>
<td>Possible, the prompt is added to the plan. When other dependencies are resolved, the job stream goes into the STUCK state, and the job remains in READY state.</td>
<td>Not possible</td>
<td>Data import creates a dummy prompt with the text &quot;The prompt prompt referenced as a dependency by job stream job_stream does not exist. A dummy prompt is added and the migrated job stream is modified to refer to it&quot;.</td>
</tr>
<tr>
<td>A dependency refers to a resource in a job or a job stream that does not exist.</td>
<td>Possible, the resource is added to the plan. When other dependencies are resolved, the job stream goes into the READY state, and the job remains in HOLD state.</td>
<td>Not possible.</td>
<td>Data import creates the resource with the same name and an availability of 0. If workstation to which the resource belongs does not exist it is associated to the MASTER workstation, and a warning is issued. The description is &quot;Resource created during initial upgrade&quot;.</td>
</tr>
<tr>
<td>A file dependency refers to a workstation that does not exist.</td>
<td>Possible, the file is added to the plan. When other dependencies are resolved, the job stream goes into the READY state, and the job remains in HOLD state.</td>
<td>Not possible.</td>
<td>Data import issues a warning and assign the file dependency to the file: TWS_INITIAL_UPGRADE_DUMMY_FILE defined on the MASTER workstation.</td>
</tr>
<tr>
<td>A job or job stream refer to a parameter that does not exist.</td>
<td>Possible, the substitution is not done.</td>
<td>Possible, the substitution is not done.</td>
<td>None.</td>
</tr>
<tr>
<td>Inter-network dependencies are defined on workstations that are not extended agents.</td>
<td>Possible, the job stream is added to the plan in the ERROR state.</td>
<td>Not possible.</td>
<td>Data import creates a dummy extended agent with an nonexistent access method:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CPUNAME TWS_DUMMY_XAGENT DESCRIPTION &quot;this is a dummy xagent with a dummy method created during initial upgrade&quot; OS OTHER NODE DUMMY_NODE TCPADDR 31111 FOR MAESTRO HOST $MASTER ACCESS &quot; DUMMY_METHOD &quot; TYPE X-AGENT AUTOLINK OFF BEHINDFIREWALL OFF FULLSTATUS OFF END</td>
</tr>
</tbody>
</table>

Chapter 5. Upgrading a version 8.2 or 8.2.1 master domain manager instance 109
Inconsistencies in the imported data

Post-upgrade issues

This section describes any issues that you must resolve after the migration is complete.

Integrating with NetView

After you migrate, if you integrate with NetView, the new NetView StartUp must be merged with the old StartUp file. Migration saves the old file as StartUp.toMerge, see Appendix A, “Integration with other IBM Tivoli products,” on page 179.

Maximizing DB2 database efficiency

If you have migrated a database containing a large number of jobs or job streams, and the database support is DB2, run the dbrunstats utility to maximize the efficiency of the database.

The DB2 database has been set up to maintain itself, so there is little user maintenance to do. However, the process of importing data does not necessarily organize the database in the most efficient way. In normal running, DB2 periodically checks the database by running an internal routine. DB2 determines when this routine must be run using a default policy. Using the statistical information that DB2 discovers by running this routine, it adjusts its internal processing parameters to maximize its performance. To avoid having to wait for the default policy for running the routine, you can do it manually. The routine is imbedded in a tool called dbrunstats, which can be run to improve performance while DB2 is processing data without causing any interruption.

See Tivoli Workload Scheduler: Administration and Troubleshooting for instructions on how to run this utility.
Chapter 6. Upgrading a version 8.3 master domain manager instance

This chapter describes how to upgrade master domain managers and backup master domain managers from Tivoli Workload Scheduler version 8.3 to version 8.4. The upgrade of version 8.2 to version 8.4 is described in Chapter 5, “Upgrading a version 8.2 or 8.2.1 master domain manager instance,” on page 77.

To upgrade agents from all supported versions, refer to the procedures described in Chapter 7, “Upgrading agents,” on page 125.

This chapter is divided into the following sections:

- "Upgrading overview"
- “Deciding how to upgrade the master domain manager and backup master domain manager” on page 112
- “Preparing to Upgrade” on page 114
- “Performing a parallel upgrade” on page 114
- “Performing a direct upgrade” on page 117
- “Building the final security file for the new environment” on page 121

Upgrading overview

This section provides an overview of the upgrade of an existing version of Tivoli Workload Scheduler 8.3 instance. It is divided into the following sections:

- "Choosing how to migrate your network"
- “Component upgrade procedures”

Choosing how to migrate your network

Tivoli Workload Scheduler version 8.4 supports backward compatibility so you can decide to upgrade your network in either of the following ways:

Top-down
Upgrade the master domain manager and backup master domain manager, and then progressively upgrade the agents. In this case, many of the new functions introduced in version 8.4 become available for each agent as it is upgraded. The disadvantage is that the same functions are not available to all agents at the same time.

Bottom-up
Upgrade the agents first, and then upgrade the master domain manager and backup master domain manager. In this case, the new functions introduced in version 8.4 will not be available until the whole network has been upgraded.

Component upgrade procedures

This section lists the upgrade procedures for Tivoli Workload Scheduler version 8.3 components and where to find them in this document.

Master domain manager and backup master domain manager
To upgrade the master domain manager and backup master domain manager follow these steps in sequence:
Overview of the upgrade

1. **Choose the direct or parallel upgrade method**
   Choose between a direct or a parallel upgrade. These alternatives are described in "Deciding how to upgrade the master domain manager and backup master domain manager" on page 114.

2. **Prepare the environment for the upgrade**
   Perform a set of activities to prepare the environment for the upgrade. These activities are described in "Preparing to Upgrade" on page 114.

3. **Perform the upgrade**
   Follow the instructions in either "Performing a parallel upgrade" on page 114 or "Performing a direct upgrade" on page 117 as appropriate.

4. **Post upgrade issues**
   There are some additional configuration steps you might need to perform, depending on your existing security settings and on the type of upgrade you choose to perform. These steps are described in "Building the final security file for the new environment" on page 121.

**Domain manager, fault-tolerant agent with and without a connector**
To upgrade a domain manager, fault-tolerant agent or fault-tolerant agent with a connector, follow the procedures described in "Upgrading agents," on page 125.

**Command line client**
To upgrade a command line client, follow the procedures described in "Upgrading a command line client" on page 131.

**Job Scheduling Console**
To upgrade the Job Scheduling Console version 8.3, follow the procedures described in *Tivoli Workload Scheduler: Job Scheduling Console User’s Guide*.

**Dynamic Workload Console**
To upgrade the Dynamic Workload Console, follow the procedures described in *Tivoli Workload Scheduler: Dynamic Workload Console Installation and Troubleshooting Guide*.

---

**Deciding how to upgrade the master domain manager and backup master domain manager**

There are several factors to take into consideration before you upgrade your master domain manager and backup master domain manager. The following topic describes these factors and outlines the available upgrade scenarios.

**Master and backup domain manager upgrade scenarios**
This section provides two upgrade scenarios from which you must choose:

- "Parallel upgrade scenario - minimizing the impact on scheduling"
- "Direct Upgrade scenario - minimizing the time to upgrade" on page 113

**Parallel upgrade scenario - minimizing the impact on scheduling**
A parallel upgrade allows you to maintain the integrity of your previous master domain manager until you are confident with the new environment. The upgrade is staged and allows you to work in coexistence with your old environment.
Parallel upgrade scenario

In the parallel scenario described in the following sections, you start by upgrading your existing backup domain manager or by installing a new Tivoli Workload Scheduler version 8.4 backup domain manager. Your new or upgraded backup master domain manager then assumes the role of your old master domain manager. You then have the choice of making this new environment permanent. This makes the upgrade of your old master domain manager an optional step. Alternatively, you can upgrade and restore the old master domain manager to its original role.

This sequence of operations is designed to minimize your out-of-service time and to ensure data integrity. The parallel upgrade involves a limited number of manual steps but has the advantage of maintaining the integrity of your current environment.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Perform one of the following depending on whether or not you already have a backup master domain manager in your version 8.3 environment:</td>
<td>• Allows the coexistence of the old master with the new environment</td>
<td>• Involves some manual configuration steps</td>
</tr>
<tr>
<td>• Upgrade your current version 8.3 backup master domain manager referencing the existing version 8.3 database</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Install a fresh backup master domain manager version 8.4, that points to the existing version 8.3 database</td>
<td>• Allows you to choose a new and better performing platform for your new master domain manager.</td>
<td></td>
</tr>
<tr>
<td>2. Switch your manager to the new backup master domain manager</td>
<td>• It is a reversible process.</td>
<td></td>
</tr>
<tr>
<td>3. Activate your new configuration.</td>
<td>• Automatically updates the database schema which is fully compatible for old and new versions</td>
<td></td>
</tr>
<tr>
<td>4. Complete your security configuration by merging old and new security settings.</td>
<td>• Allows a great degree of flexibility. You can choose to not perform step 5</td>
<td></td>
</tr>
<tr>
<td>5. Optionally, upgrade your old master domain manager and resume your original configuration restoring the upgraded master.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Direct Upgrade scenario - minimizing the time to upgrade

The direct upgrade scenario allows you to upgrade your current environment quickly, reducing manual intervention. The procedure automatically upgrades your network and database information using the input you provide. The installation wizard is the simplest way of approaching this type of upgrade because it guides you through the process.
### Direct upgrade scenario

<table>
<thead>
<tr>
<th>Steps</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Unlinking the old master domain manager and stopping it</td>
<td>• Quicker and simpler than the parallel upgrade.</td>
<td>• Scheduling might be delayed for those activities involving the master domain manager</td>
</tr>
<tr>
<td>2. Upgrading the master domain manager, automatically importing the scheduling and configuration data from the previous version.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Completing the security configuration by merging old and new security settings.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. Upgrading the backup master domain manager.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Preparing to Upgrade

Before you begin the upgrade process, complete the following tasks as appropriate:

**Perform a backup of your database**
Before you begin the upgrade process, perform a backup of your current version 8.3 Tivoli Workload Scheduler database, referring to the Oracle or DB2 documentation.

**Linux kernel**
If you are upgrading in a Linux environment that uses the LD_ASSUME KERNEL=2.4.1 environment variable, upgrade to Tivoli Workload Scheduler 8.4 in a shell that also uses the LD_ASSUME_KERNEL=2.4.1 environment variable.

**Ensure your current Tivoli Workload Scheduler installations are in correct state**
When you are upgrading your current environment, make sure the software package is in the COMMIT state (not in the UNDOABLE state). If it is in the UNDOABLE state, you must accept it to change its state to COMMIT before you upgrade to version 8.4.

### Performing a parallel upgrade

This section describes how to upgrade your environment using a parallel upgrade scenario. The scenario consists of the following procedures:

- “Parallel 1: Setting up your backup master domain manager” on page 115
  - “Parallel 1 a: Installing a new version 8.4 backup master domain manager” on page 115
    or
  - “Parallel 1 b: Upgrading your current version 8.3 backup master domain manager” on page 115
- “Parallel 2: Switching the master domain manager to the new backup master” on page 115
- “Parallel 3: Activating the new configuration” on page 115
- “Parallel 4: Complete the security file configuration” on page 117
- “Parallel 5: Optionally upgrade your version 8.3 master domain manager” on page 117
Parallel 1: Setting up your backup master domain manager

This step is divided into two alternative sub steps, depending on whether or not you already have a backup master domain manager in your version 8.3 environment:

Parallel 1 a: Installing a new version 8.4 backup master domain manager

The purpose of this step is to install a fresh backup master domain manager and attach it to your current version 8.3 network.

This backup master domain manager points to your existing Tivoli Workload Scheduler database and will become your new master domain manager:

1. To install a new backup master domain manager refer to the procedures described in Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41. Specifically, refer to the procedure described in “Common steps for installing a new master domain manager or backup master” on page 44 and subsequent sections depending on whether you are using a DB2 or an Oracle database. Ensure that your new backup master domain manager points to your current Tivoli Workload Scheduler database instance.

2. Define your new backup master domain manager as a full status agent in the domain of your version 8.3 master domain manager using the composer command interface.

Parallel 1 b: Upgrading your current version 8.3 backup master domain manager

To upgrade your current backup master domain manager, follow the procedure described in “Direct 2: Upgrading the master domain manager” on page 118 using your preferred installation method.

Parallel 2: Switching the master domain manager to the new backup master

Switch to your new backup master domain manager, which now becomes your master domain manager by issuing the following command on your old version 8.3 master domain manager:

```
conman
switchmgr masterdm;new_mgr_cpu
```

where `new_mgr_cpu` is the name of the workstation of your new or upgraded backup master domain manager.

Parallel 3: Activating the new configuration

In the preceding step you have promoted your upgraded backup master domain manager to the role of master domain manager. This role remains active only until the next plan is built (for the lifetime of the current plan). To make this configuration fully operational and permanent, you must perform the following steps:

On the new master domain manager, referred to as `new_mgr_cpu`:

1. Cancel the `FINAL` schedule associated to your old master, if one exists in the plan:

```
conman
cs old_mdm_cpu#FINAL
```
Performing a parallel upgrade

where old_mdm_cpu is the workstation name of your version 8.3 master domain manager.

2. Edit the localopts file and modify the following entry as shown:

DEFAULTWS=new_mgr_cpu

where new_mgr_cpu is the workstation name of the new master. See "Setting local options" on page 150.

3. Change the workstation definition of the new master using composer:

modify cpu=new_mgr_cpu

and substitute type=fta with type=manager

4. Repeat the preceding step, this time to modify the workstation definition of the old master and substitute type=manager with type=fta.

5. You have two options: If you are planning to make your current environment permanent, choose option (a), if you plan to upgrade your old master domain manager to restore the old configuration, then choose option (b). Either option will disable scheduling of the old FINAL schedule if it existed in the database:

a. Delete the FINAL schedule associated to the old master domain manager (if present in the database) and all related jobs:

composer
del sched=old_mdm_cpu#FINAL
del job=old_mdm_cpu#CreatePostReports
del job=old_mdm_cpu#MakePlan
del job=old_mdm_cpu#SwitchPlan
del job=old_mdm_cpu#UpdateStats

b. Set the priority of the FINAL schedule associated to the old master domain manager to the value of 0 so that it will not be scheduled, using composer:

composer
modify sched=old_mdm_cpu#FINAL

and specify priority 0.

6. Optionally, add the FINAL job stream in the database for the new master domain manager:

composer add $final

7. Rebuild the plan to activate the changes to the database:

jnextplan -for 0000

8. Optionally, add the FINAL job stream for the new master domain manager to the plan:

conman sbs new_mgr_cpu#FINAL

After this step, your backup master domain manager has become the master master domain manager of the new environment. Before you proceed to the next step, decide what to do with your old master domain manager. You have three alternatives:

- Keep the new manager as the master domain manager of your version 8.4 environment and your old master domain manager as a full status agent, upgrading it to the new version. Then proceed to "Parallel 4: Complete the security file configuration" on page 117 as a final step.
- Keep the new manager as the master domain manager of your version 8.4 environment and delete the old master from your environment. Proceed to "Parallel 4: Complete the security file configuration" on page 117 before uninstalling the old master domain manager.
Performing a parallel upgrade

- Upgrade your old master domain manager and restore the original configuration in the new environment. This alternative requires you to repeat step 3 and to make the necessary adjustments to the security file after the master domain manager upgrade. Proceed to “Parallel 5: Optionally upgrade your version 8.3 master domain manager.”

Parallel 4: Complete the security file configuration

To complete the upgrade, you need to perform some additional security file configuration steps. If you had specific security settings in your old environment, these need to be merged with the new security settings before building your final security file. Perform the steps detailed in “Building the final security file for the new environment” on page 121.

Parallel 5: Optionally upgrade your version 8.3 master domain manager

Perform the following steps:

1. From the new master domain manager, unlink the old master workstation
   conman unlink old_mdm_cpu
2. Upgrade your old master domain manager to version 8.4 using the procedure described in “Performing a direct upgrade.”
3. Link the new master domain manager to the network
   conman link old_mdm_cpu
4. From the upgraded master domain manager switch the master domain manager:
   conman
   switchmgr masterdm; old_mdm_cpu
5. To restore your upgraded master domain manager to its role permanently, perform the steps outlined in “Parallel 3: Activating the new configuration” on page 115, this time for the master workstation.

Performing a direct upgrade

This section describes how to upgrade your environment using a direct scenario. It is divided into the following procedures:

1. “Direct 1: Unlinking the master domain manager from the network and stopping it”
2. “Direct 2: Upgrading the master domain manager” on page 118
3. “Direct 3: Complete the security file configuration” on page 121
4. “Direct 4: Upgrading the backup master domain manager” on page 121

Direct 1: Unlinking the master domain manager from the network and stopping it

Before commencing the upgrade, you must unlink all workstations from the master domain manager and stop it.

Follow these steps:

1. Log in as the TWSuser.
2. Unlink all workstations in the domain:
   From the Job Scheduling Console

Chapter 6. Upgrading a version 8.3 master domain manager instance 117
Performing a parallel upgrade

a. Run a Status of all Domains plan list. The Status of all Domains table opens.

b. Right-click the domain you want to unlink and select Unlink Workstations from the pop up menu.

From the command line of the master domain manager
Issue the following command:
`conman "unlink @;noask"`

3. Stop the master domain manager:

From the Job Scheduling Console
a. Select the master domain manager workstation
b. Right-click the workstation and select Stop from the pop-up menu.

From the command line of the master domain manager
Issue the following command:
`conman "stop;wait"`

4. From the command line of the master domain manager, stop the netman process as follows:
   a. On UNIX, run:
      `conman "shut"
   
      Note: Do not use the UNIX kill command to stop Tivoli Workload Scheduler processes.
   b. On Windows, run the `shutdown.cmd` command from the Tivoli Workload Scheduler home directory.

5. Verify that all services and processes are not running, as follows:
   a. On UNIX, type the command:
      `ps -u TWSuser`
      Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, JOBMAN, stageman.
   b. On Windows, type the command:
      `<drive>\unsupported\listproc.exe`
      where `<drive>` is the Tivoli Workload Scheduler home directory. Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, stageman, JOBMON, tokensrv, batchup.
   
      Also, ensure that no system programs are accessing the directory or subdirectories, including the command prompt, and that in Windows Explorer the Administrative Tools→Services panel is not open.

Direct 2: Upgrading the master domain manager
This section describes how to upgrade a master domain manager.

You can upgrade a master domain manager using the following installation methods:

- "Upgrading a master domain manager or backup master using the installation wizard" on page 119
- "Upgrading a master domain manager or backup master domain manager using the silent installation" on page 121
Upgrading a master domain manager or backup master using the installation wizard

To upgrade a Tivoli Workload Scheduler master domain manager from version 8.3 perform the following steps.

1. Insert the installation CD according to the operating system. See "The installation CDs" on page 21.

2. Run setup for the operating system on which you are upgrading:
   - On Windows operating systems, WINDOWS\SETUP.exe
   - On UNIX and Linux operating systems, SETUP.bin.
   - Alternatively, start the launchpad as follows and select the Tivoli Workload Scheduler installation:

   **On Windows systems**
   - From the root directory of the CD run the launchpad.exe file.

   **On UNIX systems**
   - From the root directory of the CD run the launchpad.sh file.

3. The installation wizard is launched. Select the installation wizard language. Click OK.

4. Read the welcome information and click Next.

5. Read and accept the license agreement. Click Next.

6. Select the instance of Tivoli Workload Scheduler from the drop-down list. The Upgrade an instance of Tivoli Workload Scheduler button is automatically selected. Click Next.

7. Specify the following information:

   **Back up the previous Tivoli Workload Scheduler instance**
   - Select whether to back up the previous instance.

   **Backup Destination Directory**
   - When you select to back up the previous instance, specify the directory where the backup is to be located.

   **Backup profile destination directory**
   - The upgrade procedure performs a backup of your WebSphere Application Server 6.0.2 (WAS) profile. Your current settings are transferred to the new Websphere Application Server automatically. A default backup path is provided for you. If you want to save the profile to a different path, specify it in this field. Click Next.

8. Enter the following information:
   - Type the password of the Tivoli Workload Scheduler user for which you are upgrading the instance.
   - If you have changed the WebSphere Application Server (WAS) authentication user name and password from your previous installation, enter the new user name and password. If you have not changed these Websphere Application Server values in your current installation, leave these fields blank.

   Click Next

9. Specify the following application server port information. For an explanation on the ports used by Websphere Application Server, see Chapter 3, “Installing a fresh Tivoli Workload Scheduler,” on page 41. Accept the default values unless you know that the default ports are already in use. If you need to provide different values, any valid port number can be used:

---

Chapter 6. Upgrading a version 8.3 master domain manager instance  119
Performing a parallel upgrade

| SAS Port         | The default value is 31119. |
| Server Authentication Port | The default value is 31120. |
| Server Mutual Authentication Port | The default value is 31121. |
| ORB Port         | The default value is 31122. |
| Administration Port | The default value is 31123. |
| Administration Secure Port | The default value is 31124. |

**Event Processor Monitoring (EIF) Port**
This port is used by the event management feature. The default value is 31131. This port is not requested for backup master domain managers.

The procedure checks the availability of the specified ports, which might take a few seconds. Click Next.

10. Specify the installation path of your RDBMS instance. Click Next.
11. Depending on the type of RDBMS you are using, specify the following information:
   For DB2 version 8.2:
   **DB2 Server administrator user**
   The user name of the administrator of the DB2 server instance. This user can also be any user having SYSADM or SYSCTRL authority on the DB2 server. On UNIX, verify that you are able to switch to this user and that it can load the DB2 environment.

   **DB2 Server administrator password**
   The password of the DB2 server administrator user, or of the user with SYSADM or SYSCTRL authority. You are asked to confirm the password.

   If you have a DB2 Enterprise client installed on a UNIX platform, specify also the **DB2 local client user name**.

   For Oracle:
   **Oracle Administrator User**
   The name of the Oracle Administrator user

   **Oracle Administrator user password**
   The password of the Oracle Administrator user. You are asked to confirm the password.

   Click Next.
12. Specify the following information needed to update the Tivoli Workload Scheduler database:
   For DB2:
   **Report tablespace name**
   The name of the tablespace used to store event logs
Performing a parallel upgrade

Report tablespace path
The path of the tablespace used to store event logs.

For Oracle:

Tivoli Workload Scheduler Oracle user password
The password of the Tivoli Workload Scheduler Oracle user

Tivoli Workload Scheduler report tablespace
The path of the Oracle tablespace for Tivoli Workload Scheduler reports.

13. Review the summary data. If you need to change some values, click Back. Before clicking Next to start the upgrade process, note that the installation process is performed as a series of steps. If a step fails, you might be able to correct the problem and resume the installation. If the installation fails, before taking any action, refer to the Administration and Troubleshooting manual to understand how to proceed.

14. When the installation completes successfully, an appropriate panel is displayed.

15. Click Finish.

Upgrading a master domain manager or backup master domain manager using the silent installation
To upgrade your Tivoli Workload Scheduler master domain manager or backup master domain manager instance, use one of the following response files:
- TWS84_UPGRADE_MDM.txt
- TWS84_UPGRADE_BACKUP_MDM.txt

and follow the procedure described in "Performing a silent installation” on page 60.

Direct 3: Complete the security file configuration
To complete the upgrade, you need to merge your current security file settings with the new security settings for version 8.4 before you build your final security file. You might need to perform one or more of the steps detailed in “Building the final security file for the new environment.”

Direct 4: Upgrading the backup master domain manager
Upgrade your version 8.3 backup master domain manager, following the steps outlined in “Direct 2: Upgrading the master domain manager” on page 118 using your preferred installation method.

Building the final security file for the new environment
The upgrade scenarios do not affect your 8.3 security file which is left unchanged from your old installation. Version 8.4 introduces new security statements for the event management and reporting features. If you have specific security settings in your 8.3 environment, these settings must be manually merged with the new settings before you build the final security file for your 8.4 environment. The statements you add manually might vary depending on your security settings and on whether you have chosen the parallel or direct upgrade scenario, as explained in the following sections.

Parallel upgrade with a new backup master domain manager installation
If you installed a new version 8.4 backup master domain manager by performing “Parallel 1 a: Installing a new version 8.4 backup master manager”
Performing a parallel upgrade

If you had specific user security settings in your old version 8.3 environment, these settings must be merged with the new settings before you build your final security file. Perform the following steps:

1. On the old version 8.3 master domain manager, log in as the TWS_user and set the Tivoli Workload Scheduler environment.
2. Use the dumpsec command to extract your version 8.3 security settings to a text file:
   
   `dumpsec > v83_sec_file`

   where `v83_sec_file` is the text file created by the dumpsec command. Transfer this file to your new version 8.4 master domain manager system.
3. Edit the `v83_sec_file` and add the following new event management and reporting statements:

   ```
   REPORT
   NAME= TWS_user ACCESS=DISPLAY
   EVENTRULE
   NAME= TWS_user ACCESS=ADD,DELETE,DISPLAY,MODIFY,LIST,UNLOCK
   ACTION
   PROVIDER= TWS_user ACCESS=DISPLAY,SUBMIT,USE,LIST
   EVENT
   PROVIDER= TWS_user ACCESS=USE
   ```

   These statements can also be copied from the new security file on the master domain manager system. This is done by performing step 2 on the new security file to extract the security settings. These statements can be copied and pasted from the new security file to the `v83_sec_file`. Save your changes to the `v83_sec_file`.
4. Build your final security file for your version 8.4 master domain manager using the makesec command:

   ```
   makesec v83_sec_file
   ```

5. Enable the Event Processor Monitoring port (EIF port) which is set to 0 by default. Issue the following optman command on your new master domain manager:

   ```
   optman chg ee <new_port_value>
   ```

   where `<new_port_value>` is any valid free port number on the new master domain manager. For more information on the optman command, see "Setting global options" on page 143.

If you did not have customized user security settings in your old environment, you already have the updated settings in your new security file, and no other activity is required on this file. Run step 5 to enable the Event Processor Monitoring Port as a final step, because this port is disabled by default on backup master domain managers.

### Parallel upgrade with an upgraded backup master domain manager

If you upgraded your existing version 8.3 backup master domain manager performing "Parallel 1 b: Upgrading your current version 8.3 backup master domain manager" on page 115, your version 8.3 security file is unchanged. You need to add the version 8.4 settings manually. On your upgraded backup master domain manager perform the steps described in 121.

### Parallel upgrade - if you restored your original master domain manager

If you decided to upgrade your old master domain manager and to restore
Performing a parallel upgrade

its role in the new configuration, you need to add the new security settings by performing the steps described in 121 with the exception of step 5.

Direct Upgrade

After a direct upgrade, you need to add the new security settings manually by performing the steps described in 121 with the exception of step 5.
Performing a parallel upgrade
Chapter 7. Upgrading agents

This section describes how to upgrade agents. Before you upgrade agents ensure you performed the procedure described in “Unlinking and stopping Tivoli Workload Scheduler when upgrading agent workstations.” When the upgrade procedure is successful it is not possible to roll back to the previous version. Rollback is only possible for upgrades that fail. You can upgrade agents using the following installation methods:

- “Upgrading agents using the installation wizard” on page 126
- “Upgrading agents using a silent installation” on page 128
- “Upgrading agents using the twsinst script” on page 128
- “Upgrading agents using Software Distribution” on page 130
- “Upgrading a command line client” on page 131

Unlinking and stopping Tivoli Workload Scheduler when upgrading agent workstations

Before you perform an upgrade on an agent workstation ensure that all Tivoli Workload Scheduler processes and services are stopped. If you have jobs that are currently running, the related processes must be stopped manually. Follow these steps:

1. From the Job Scheduling Console, unlink the target workstation from the other workstations in the network. Otherwise, from the command line of the master domain manager, use the following command:
   ```
   conman "unlink workstationname;noask"
   ```
2. From the Job Scheduling Console, stop the target workstation. Otherwise, from the command line of the master domain manager, while logged in as the TWSuser, use the following command:
   ```
   conman "stop workstationname;wait"
   ```
3. From the command line (UNIX) or command prompt (Windows), stop the netman process as follows:
   - On UNIX, run:
     ```
     conman "shut workstationname"
     ```
   - On Windows, run the shutdown.cmd command from the Tivoli Workload Scheduler home directory.

   **Note:** Do not use the UNIX kill command to stop Tivoli Workload Scheduler processes.

4. If you are updating an agent, remove (unmount) any NTFS mounted directories from the master domain manager.
5. If you are upgrading an installation that includes the connector, stop the connector.

To verify if any services and processes are still running, complete the following steps:

- On UNIX, type the command:
  ```
  ps -u TWSuser
  ```
Unlinking and stopping Tivoli Workload Scheduler

Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, JOBMAN, stageman.

- On Windows, type the command:
  \(<drive>\)unsupported\listproc.exe

Verify that the following processes are not running: netman, mailman, batchman, writer, jobman, stageman, JOBMON, tokensrv, batchup.

Also, ensure that no system programs are accessing the directory or subdirectories, including the command prompt, and that in Windows Explorer the Administrative Tools\Services panel is not opened.

Notes:
1. If you are upgrading in a Windows environment the Tivoli Token Server must be running.
2. Before you upgrade, make sure that the conman command line is not running.

Upgrading agents using the installation wizard

To upgrade an agent using the installation wizard, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.
2. Run setup for the operating system on which you are upgrading:
   - On Windows operating systems, WINDOWS\SETUP.exe
   - On UNIX and Linux operating systems, SETUP.bin.
   - Alternatively, start the launchpad as follows and select the Tivoli Workload Scheduler installation:

   **On Windows systems**
   - from the root directory of the CD run the launchpad.exe file.

   **On UNIX systems**
   - From the root directory of the CD run the launchpad.sh file.
3. The installation wizard is launched. Select the installation wizard language. Click OK.
4. Read the welcome information and click Next.
5. Read and accept the license agreement. Click Next.
6. Select an existing installation of a previous release of the product from the drop down list. The instance can be identified by its group name. Click Next.
7. Review the Tivoli Workload Scheduler user data. Click Next.
8. Review the summary data and click Next.
9. When the installation completes successfully, a successful completion panel is displayed. If the upgrade ended with errors, see Administration and Troubleshooting for information on how to correct the error.
10. Click Finish.

Upgrading an agent with connector using the installation wizard

To upgrade a Tivoli Workload Scheduler version 8.3 agent with a connector using the installation wizard, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.
2. Run setup for the operating system on which you are upgrading:
   - On Windows operating systems, WINDOWS\setup.exe
   - On UNIX and Linux operating systems, setup.bin
   - Alternatively, start the launchpad as follows and select the Tivoli Workload Scheduler installation:
     - **On Windows systems**
       - from the root directory of the CD run the launchpad.exe file.
     - **On UNIX systems**
       - From the root directory of the CD run the launchpad.sh file.

3. The installation wizard is launched. Select the installation wizard language. Click OK.

4. Read the welcome information and click Next.

5. Read and accept the license agreement. Click Next.

6. Select the instance of the agent version 8.3 with connector from the drop-down list. The Upgrade an instance of Tivoli Workload Scheduler button is automatically selected. Click Next.

7. Specify the backup profile destination directory. This information is needed by the Tivoli Workload Scheduler upgrade procedure to perform a backup of your WebSphere Application Server (WAS) profile. Your current settings are transferred to Websphere Application Server automatically. Click Next.

8. Enter the password of the Tivoli Workload Scheduler user for which you are upgrading the agent and connector instance. If you changed the WebSphere Application Server authentication user name and password from your previous installation, you must supply them here. If you did not change these values in your installation, leave these fields blank. Click Next.

9. Specify the values for the following ports used by the embedded version of the WebSphere Application Server. Default values are presented to you:

   - **SAS Port**
     - The default value is 31119.

   - **Server Authentication Port**
     - The default value is 31120.

   - **Server Mutual Authentication Port**
     - The default value is 31121.

   - **ORB Port**
     - The default value is 31122.

   - **Administration Port**
     - The default value is 31123.

   - **Administration Secure Port**
     - The default value is 31124.

10. Review the summary data and click Next.

11. The installation result is displayed. If the upgrade is unsuccessful, refer to Administration and Troubleshooting.

12. Click Finish.
**Upgrading agents using a silent installation**

To upgrade an agent using a silent installation, follow the procedure described in "Performing a silent installation" on page 60 with the appropriate response files:
- TWS84_UPGRADE_Agent.txt for a Tivoli Workload Scheduler 8.2.x or 8.3 agent
- TWS84_UPGRADE_Connector.txt for a Tivoli Workload Scheduler 8.3 agent with connector

**Upgrading agents using the twsinst script**

Use this procedure to upgrade on all supported UNIX operating systems. This procedure uses the command line `twsinst` script to upgrade. Refer to "Supported operating systems" on page 27 for a list of supported operating systems and requirements.

To upgrade agents using the `twsinst` script, perform the following steps:

1. Insert the installation CD according to the operating system. See "The installation CDs" on page 21.
2. Log in as root, and change your directory to `TWShome`.
3. Locate the directory of the operating system where you want to install, and run the `twsinst` script as follows:

   ```bash
   twsinst -update -uname username
   [-inst_dir install_dir]
   [-backup_dir backup_dir]
   [-nobackup_dir]
   [-lang lang-id]
   [-create_link]
   [-skip_usercheck]
   [-reset_perm]
   ```

   `-update` upgrades an existing agent installation.

   `-uname` The name of the user for which Tivoli Workload Scheduler is being updated. The software is updated in this user’s home directory. This user name is not to be confused with the user performing the upgrade.

   `-inst_dir` The directory of the Tivoli Workload Scheduler installation. This path cannot contain blanks. If not specified, the path is set to the `username` home directory.

   `-backup_dir` An alternative directory (which must be created manually) as the destination for the backup copy of a previous version.

   If you do not specify this option when running an upgrade, the following default value is used:

   ```bash
   $BACKUP_DIR = $INST_DIR_backup_$TWS_USER
   ```

   where:
   - `$INST_DIR` is the installation path (the user home directory on UNIX and Linux).
   - `$TWS_USER` is the user name.

   For example:
$INST_DIR=/opt/TWS/TWS82
$TWS_USER=user82
$BACKUP_DIR=/opt/TWS/TWS82_backup_user82
$BACKUP_SUBDIR=/opt/TWS/TWS82_backup_user82/TWS82

In the backup directory you must also create a subdirectory to include as the latest directory of the installation path.

-nobackup_dir
No backup is made.

-lang
The language in which the twsinst messages are displayed. If not specified, the system LANG is used. If the related catalog is missing, the default C language catalog is used.

Note: The -lang option does not relate to the supported language packs. By default, all supported language packs are installed when you install using the twsinst script.

-create_link
Create the symlink between /usr/bin/mat and TWS/home/bin/at. See Table 9 on page 23 for more information.

-skip_usercheck
Skip the check of the user in the /etc/password file or using the su command. Enable this option if the authentication process within your organization is not standard, thereby disabling the default authentication option.

-reset_perm
Reset the permissions of the libatrc library.

For example, a sample twsinst script to upgrade a version 8.2 fault-tolerant agent to a version 8.4 agent workstation:
./twsinst -update -uname twuser

To roll back to a previous instance, run the following command:
twsinst -restore -uname username [-skip_usercheck]

-restore
If the installation fails, restores the backup automatically created by Configuration Manager for the user specified in uname.

-uname
The name of the user for which Tivoli Workload Scheduler is being updated or rolled back. The software is updated in this user’s home directory. This user name is not to be confused with the user performing the installation logged on as root.

-skip_usercheck
Skip the check of the user in the /etc/password file or using the su command. Enable this option if the authentication process within your organization is not standard, thereby disabling the default authentication option.
Upgrading using Software Distribution

Upgrading agents using Software Distribution

A number of Tivoli Workload Scheduler parameters are used by the software package block to perform the upgrade. You can assign values to each variable to reflect the installation that is being upgraded, otherwise the default value is assigned.

When you upgrade agents using Software Distribution, the following variables are required:

- `install_dir`
- `tws_user`
- `pwd`
- `fresh_install`
- `upgrade`
- `from_release`

For a description of the variables, see Table 21 on page 64.

To perform the upgrade, complete the following steps:

1. Create a software package profile that has the following name:
   
   `FP_TWS_operating_system_TWSuser.8.4.00`
   
   where: `operating_system` is the operating system where you are installing and `TWSuser` is the user of the installation.
   
   When you import the software package block, you must pass the name of the profile to `wimpspo` so that the Configuration Manager endpoint catalogs the name correctly.

2. Import the software package block using the `wimpspo` command.

3. Install the software package block using the `winstsp` command.

   **Note:** When upgrading using the `winstsp` command, make sure to specify the `install_dir` variable. If you installed the previous version in a directory other than the default and you do not specify `install_dir`, Tivoli Workload Scheduler is installed as a fresh installation.


**Example**

The following is an example of the settings required to upgrade a Tivoli Workload Scheduler, version 8.2 fault-tolerant agent to Tivoli Workload Scheduler, version 8.4 on a Windows workstation. See Table 21 on page 64 for a description of the parameters.

```
  winstsp -D install_dir="d:\Program Files\IBM\TWS\juno" -D tws_user="juno"
  -D this_cpu="saturn" -D tcp_port="31111" -
  D fresh_install="false" -D upgrade="true" -D from_release="8.2" -D backup="true"
  FP_TWS_WINDOWS_juno.8.4.00 TWSep
```
Upgrading a command line client

To upgrade a Tivoli Workload Scheduler 8.3 command line client using the installation wizard, perform the following steps:

1. Insert the installation CD according to the operating system. See “The installation CDs” on page 21.
2. Run setup for the operating system on which you are upgrading:
   - On Windows operating systems, WINDOWS\SETUP.exe
   - On UNIX and Linux operating systems, SETUP.bin
   - Alternatively, start the launchpad as follows and select the Tivoli Workload Scheduler installation:
     - **On Windows systems**
       - from the root directory of the CD run the launchpad.exe file.
     - **On UNIX systems**
       - From the root directory of the CD run the launchpad.sh file.
3. The installation wizard is launched. Select the installation wizard language. Click OK.
4. Read the welcome information and click Next.
5. Read and accept the license agreement. Click Next.
6. Select the instance of the version 8.3 command line client from the drop-down list. The Upgrade an instance of Tivoli Workload Scheduler button is automatically selected. Click Next.
7. Review the information displayed on the summary panel. Click Next.
8. When the installation completes, click Finish.

To upgrade a command line client using the silent installation, follow the procedure described in “Performing a silent installation” on page 60 using the TWS84_UPGRADE_CLI.txt response file.
Chapter 8. Uninstalling Tivoli Workload Scheduler

This chapter describes how you uninstall Tivoli Workload Scheduler. It is divided into the following sections:

- “Uninstalling using the uninstallation wizard”
- “Performing a silent uninstallation” on page 134
- “Uninstalling agents using the twsinst script” on page 134
- “Uninstalling using the Software Distribution CLI” on page 135
- “Uninstalling a command-line client” on page 135

The uninstaller program is created during the install procedure. Wherever possible, use the same method you chose to install the product when uninstalling the product. For example, if you installed the product using the installation wizard, use the uninstaller program to subsequently remove the product.

Uninstalling the product does not remove files created after Tivoli Workload Scheduler was installed, nor files that are open at the time of uninstall. If you do not need those files, you have to remove them manually. If you intend to reinstall and need to use the files, make a backup before starting the installation process. The uninstallation does not remove your DB2 or Oracle database.

Notes:
1. The Tivoli Workload Scheduler engine is a prerequisite for other products and features you can install, such as Tivoli Workload Scheduler for Applications and the connector. Before you uninstall the engine, uninstall all the additional features.
2. Refer to Tivoli Workload Scheduler Administration and Troubleshooting for information about removing Tivoli Workload Scheduler manually.

Uninstalling using the uninstallation wizard

The uninstaller program removes product files, registry keys, and services. It removes the binaries related to the Tivoli Workload Scheduler agent installed, the distributed connector, and the language packs.

To uninstall Tivoli Workload Scheduler, perform the following steps:
1. Ensure that all Tivoli Workload Scheduler processes and services are stopped, and that there are no active or pending jobs.
2. Navigate to the TWS_home/uninstall path.
3. Run the uninstall script:
   - On Windows operating systems:
     uninstaller.exe
   - On UNIX and Linux operating systems:
     ./uninstall.bin
4. Select the Tivoli Workload Scheduler instance you want to uninstall:
   - If you are uninstalling a master domain manager, the wizard removes the selected instance and any additional feature installed for that instance.
   - If you are uninstalling an agent, you can choose if you want to uninstall the connector only, or both the agent and connector simultaneously.
Uninstalling using the uninstallation wizard

For UNIX and Linux operating systems, uninstall is performed in the language of the locale and not the language set during the installation phase. If you want to uninstall agents in a language other than the locale of the machine, use the twsinst script as follows:

<TWShome>/twinst -uninst -uname <user_name> -lang<language>

where language is the language of the uninstallation. For a detailed description of all the available parameters for the twsinst script, see "Uninstalling agents using the twsinst script."

Performing a silent uninstallation

For a silent uninstallation of a Tivoli Workload Scheduler master domain manager or backup master domain manager perform the following steps:

1. Ensure that all Tivoli Workload Scheduler processes and services are stopped, and that there are no active or pending jobs.
2. Navigate to the TWShome/_uninstall path.
3. Enter the following command:
   • On Windows operating systems:
     uninstaller.exe -silent
   • On UNIX and Linux operating systems:
     ./uninstall.bin -silent

For a silent uninstallation of an agent, a connector, or both, perform the following steps:

1. Ensure that all Tivoli Workload Scheduler processes and services are stopped, and that there are no active or pending jobs.
2. Copy the TWS84_UNINSTALL_Agent.txt response file from the installation CD in the \RESPONSEFILES\ directory to a local directory and edit it as appropriate.
3. Save the file with your changes.
4. Navigate to the TWShome/_uninstall path.
5. Enter the following command:
   • On Windows operating systems,
     SETUP.exe -options <local_dir>\TWS84_UNINSTALL_Agent.txt -silent
   • On UNIX and Linux operating systems,
     ./SETUP.bin -options <local_dir>/TWS84_UNINSTALL_Agent.txt -silent

Note: If you intend to reinstall after performing a silent uninstallation, you must first close and reopen the shell to correctly reset the environment variables.

Uninstalling agents using the twsinst script

Follow these steps to uninstall Tivoli Workload Scheduler agents using the twsinst script.

1. Before uninstalling, stop any existing Tivoli Workload Scheduler processes that were created on this particular system. If you have jobs that are currently running, the related processes must be stopped manually. For information about stopping the processes and services see Tivoli Workload Scheduler Administration and Troubleshooting.
2. Log in as root and change your directory to TWShome
3. Run the twsinst script as follows:
Uninstalling using the twsinst script

```bash
twsinst -uninst -uname <username>
    [-lang <lang_id>]
    [-skip_usercheck]
```

- **-uninst**
  Uninstalls Tivoli Workload Scheduler.

- **-uname <username>**
  The name of the user for which Tivoli Workload Scheduler is uninstalled. This user name is not to be confused with the user performing the uninstallation logged on as root.

- **-lang <lang_id>**
  The language in which the twsinst messages are displayed. If not specified, the system LANG is used. If the related catalog is missing, the default C language catalog is used.

  **Note:** The `-lang` option is not to be confused with the Tivoli Workload Scheduler supported language packs.

- **-skip_usercheck**
  Skip the check of the user in the `/etc/password` file or using the `su` command.

For example, a sample `twsinst` script that uninstalls the Tivoli Workload Scheduler agent, originally installed for user named twsuser:

```
./twsinst -uninst -uname twsuser
```

Uninstalling using the Software Distribution CLI

You can uninstall Tivoli Workload Scheduler using the Software Distribution command `wremovsp` Before uninstalling, you must stop all Tivoli Workload Scheduler processes using the `conman shutdown` command.

For example, to uninstall on Windows, use the following command:

```
wremovsp Tivoli_TWS_WINDOWS.spb [subscribers...]```

The software package block that installs language packs can also be removed in this way. Refer to *Tivoli Workload Scheduler Administration and Troubleshooting* for information about removing Tivoli Workload Scheduler manually.

Uninstalling a command-line client

You can uninstall a command line client using the uninstallation wizard or by performing a silent uninstallation. To uninstall a command-line client perform the following steps:

1. Navigate to the `CLI_home/_uninstall` path, where `CLI_home` is the installation path of your command-line client.

2. For an uninstallation using the wizard:
   - On Windows operating systems:
     ```
     uninstaller.exe
     ```
   - On UNIX and Linux operating systems:
     ```
     ./uninstall.bin
     ```
   
   OR

3. For a silent uninstallation:
   - On Windows operating systems:
Uninstalling using the Software Distribution CLI

- uninstaller.exe -silent
- On UNIX and Linux operating systems:
  
  ./uninstall.bin -silent
Part 3. Configuring
Chapter 9. Configuring after installation

This chapter describes configuration tasks that might be required at the end of the installation procedure you followed. Each procedure points you to the appropriate tasks. It consists of the following sections:

- “Checking the Netman process”
- “Setting the environment variables”
- “Configuring a master domain manager” on page 140
- “Configuring a backup master domain manager” on page 140
- “Configuring an agent” on page 141

Checking the Netman process

The Netman process is automatically started at the end of installation. This is to verify that the installation process ended successfully. For more information on Tivoli Workload Scheduler processes, see Tivoli Workload Scheduler: Reference Guide

Setting the environment variables

Before you start configuring your Tivoli Workload Scheduler components, set the environment variables.

On Windows operating systems, run the tws_env.cmd shell script to set up both the PATH and TWS_TISDIR variables. For example, if Tivoli Workload Scheduler is installed in the C:\Program Files\IBM\TWS\jdoe directory, the PATH variable is set as follows:

C:\Program Files\IBM\TWS\jdoe;c:\Program Files\IBM\TWS\jdoe\bin

Note: If you have more than one version of Tivoli Workload Scheduler installed on your computer, make sure TWS_TISDIR points to the latest one. This ensures that the most recent character set conversion tables are used.

On UNIX and Linux operating systems, run the tws_env shell script to set up both the PATH and TWS_TISDIR variables. For example, if Tivoli Workload Scheduler is installed in the /opt/maestro directory, tws_env sets the PATH variable as follows:

PATH=/opt/maestro:/opt/maestro/bin:$PATH
export PATH

and the TWS_TISDIR variable to TWShome. For example:

TWS_TISDIR=/opt/maestro
export TWS_TISDIR

The tws_env script has two versions:

- tws_env.sh for Bourne and Korn shell environments
- tws_env.csh for C Shell environments
Configuring a master domain manager

After you have installed a master domain manager, if you did not select to automatically add the final job stream during installation and want to do so, follow the steps in this section to add the final job stream to the database and run JnextPlan. This job stream is placed in production every day and runs JnextPlan prior to the start of a new day. The installation creates the $final file in the TWShome directory on your workstation containing the final job stream definition. You can use $final or create and customize a new file. See Tivoli Workload Scheduler: Reference Guide for details about customizing the final job stream.

The following is an example of how to configure a master domain manager after installation:

1. Log in as TWSuser.
2. Set the environment variables. See “Setting the environment variables” on page 139.
3. Run the composer command.
4. Add the final job stream definition to the database by running the following command:
   ```
   add $final
   ```
   where $final is the name of the file containing the definition of the Final job stream.
5. Exit the composer command line.
6. Run the JnextPlan job:
   ```
   JnextPlan
   ```
   You can automate this step after installation. See Tivoli Workload Scheduler: Reference Guide.
7. When JnextPlan completes, check the status of Tivoli Workload Scheduler:
   ```
   conman status
   ```
   If Tivoli Workload Scheduler started correctly the status returned by the command is Batchman=LIVES.
8. Raise the workstation limit value to allow jobs to run. The default job limit after installation is 0, so no jobs run. Raise the job limit to allow jobs to run, for example to 10 jobs:
   ```
   conman "limit ;10"
   ```
   If no workstation name is specified for the limit command, the default value is the current login workstation.

Configuring a backup master domain manager

After you have installed a backup master domain manager, perform the following configuration steps:

1. Login as TWSuser on your master domain manager
2. Add the backup master username and password to the useropts file. See Tivoli Workload Scheduler: Reference Guide.
3. Set the environment variables by running `tws_env`
4. Define the backup master as a full status autolink fault-tolerant agent in the Tivoli Workload Scheduler database, using the `composer` command interface or the Job Scheduling Console. In this example using `composer`:

```
composer new
```

5. Type the workstation definition in the text editor, for example:

```
cpuname DM1
os UNIX
node domain1
description "Fault-tolerant Agent"
for Maestro
type FTA
autolink on
end
```

For more information on workstation definitions, refer to the *Tivoli Workload Scheduler Reference Guide.*

6. Run JnextPlan `-for 0000` to include the backup master workstation in the plan and to send the Symphony file to it.

7. Change the workstation limit to allow jobs to run on the workstation. For example, set the number of jobs to run concurrently on the workstation to 10:

```
composer "limit DM1;10
```

---

**Configuring an agent**

After you have installed an agent define the workstation in the database and link the workstation from the master. You can perform this task using the Job Scheduling Console or the command line interface. Refer to the *Tivoli Workload Scheduler Job Scheduling Console User’s Guide* for information. The following is an example of configuring an agent after installation using the command line interface:

1. Login to the master domain manager as `TWSuser`.
2. Set the environment variables by running `tws_env`.
3. Create the workstation definition in the Tivoli Workload Scheduler database. Open a command line window and enter the following commands:

```
composer
new
```

4. Type the workstation definition in the text editor. For example:

```
CPUNAME F235007_00
   DESCRIPTION "Fault tolerant agent"
   OS UNIX
   NODE lab235007
   TCPADDR 31111
   DOMAIN MASTERDM
   FOR MAESTRO
   TYPE FTA
   AUTOLINK ON
   BEHINDFIREWALL OFF
   FULLSTATUS OFF
END
```

Run JnextPlan with the option `-for 0000` to add the agent workstation definition to the plan and to send the Symphony file to it. For more information on workstation definitions, refer to the *Tivoli Workload Scheduler Reference Guide.*
5. If you set the autolink parameter to OFF, issue the link command from the master domain manager to link the agent and to download the Symphony file to it:
   conman "link ftoname"

6. Change the workstation limit to allow jobs to run on the workstation. For example, set the number of jobs to run concurrently on the workstation to 10:
   composer "limit F235007_00;10"
Chapter 10. Performing optional customization

After installing the product, you want to customize it to fit your operational requirements. This chapter describes optional customization steps for your Tivoli Workload Scheduler installation. It is divided into the following sections:

- “Setting global options”
- “Setting local options” on page 150
- “Tivoli Workload Scheduler console messages and prompts” on page 161
- “Enabling the time zone feature” on page 162

Note: For information about automating the production cycle and managing the production environment, see the Reference Guide.

Setting global options

You set global options using the optman command. Note that you need to have the following security permissions to the global options file in the Tivoli Workload Scheduler security file to work with this command; that is:

- For optman ls or optman show:
  FILE NAME=GLOBALOPTS ACCESS=DISPLAY
- For optman chg:
  FILE NAME=GLOBALOPTS ACCESS=MODIFY


To view the values of the global options, use the following command:

optman ls

Note: Most of the options need JnextPlan to be run before they take effect. Some of these options require a restart of the Websphere Application Server. These actions are indicated in the option description. See Tivoli Workload Scheduler: Reference Guide for more information on the JnextPlan command.

To show information about an option, use the following command:

optman show {optionName | optionShortName}

where optionName or optionShortName are the options you want to view.

To change the values of an option, use the following command:

optman chg {optionName | optionShortName} = value

where optionName or optionShortName can be:

- baseRecPrompt | bp
  Specify the maximum number of prompts that can be displayed to the operator after a job abends. The default value is 1000. Run JnextPlan to make this change effective.

- carryStates | cs
  Carry job states is a preproduction option that affects the operation
Setting global options with the optman command

of the stageman command. Its setting determines the jobs, by state, to be included in job streams that are carried forward. Enclose the job states in parentheses, double quotation marks, or single quotation marks. The commas can be replaced by spaces. The valid internal job states are as follows:

<table>
<thead>
<tr>
<th>abend</th>
<th>abep</th>
<th>add</th>
<th>done</th>
<th>exec</th>
<th>fail</th>
</tr>
</thead>
<tbody>
<tr>
<td>hold</td>
<td>intro</td>
<td>pend</td>
<td>ready</td>
<td>rjob</td>
<td>sched</td>
</tr>
<tr>
<td>skel</td>
<td>succ</td>
<td>succp</td>
<td>susp</td>
<td>wait</td>
<td>waitd</td>
</tr>
</tbody>
</table>

Some examples of the option are as follows:

carryStates="abend,exec,hold,intro"
carryStates='abend,exec,hold,intro'
carryStates="abend, exec, hold, intro"
carryStates='abend, exec, hold, intro'
an empty list is entered as follows:

carryStates=null

The default value is null. Run JnextPlan to make this change effective.

cOMPANYNAME | cn

This is your company’s name. The maximum length is 40 characters. If the name contains spaces, enclose it in quotation marks ("). If you use the Japanese-Katakana language set, always enclose the name within single or double quotes. Run JnextPlan to make this change effective.

deploymentFrequency | df

Used in event rule management. Specify in minutes the frequency with which rules are to be checked to detect if there are changes to deploy. All active rules (they have the isDraft property set to NO in their definition) that have been changed or added since the last deployment are deployed.

Valid values are in the 0-60 minutes range. If you specify 0, the changes are not deployed automatically and you must use the planman deploy command. The default value is 5 minutes. The change is effective immediately.

enCarryForward | cf

Carry forward is a preproduction option that affects the operation of the stageman command. Its setting determines whether or not job streams that did not complete are carried forward from the old to the new production plan (Symphony). Enter yes to have uncompleted job streams carried forward only if the Carry Forward option is enabled in the job stream definition. Enter all to have all uncompleted job streams carried forward, regardless of the Carry Forward option. Enter no to completely disable the carry forward function. When the stageman -carryforward command is used it overridesenCarryForward. See Tivoli Workload Scheduler: Reference Guide for more information. The default value is all. Run JnextPlan to make this change effective.

enCentSec | ts

Defines how the security file is used within the network. Centralized security is not relevant to an end-to-end scheduling environment.
Setting global options with the optman command

If it is set to **yes**, the security files of all the workstations of the network can be created and modified only on the master, and the Tivoli Workload Scheduler administrator is responsible for their production, maintenance, and distribution.

If it is set to **no**, the security file of each workstation can be managed by its root user or administrator. The local user can run the `makesec` command to create or update the file.

The default value is **no**. Run `JnextPlan` to make this change effective. See *Tivoli Workload Scheduler: Reference Guide* for more information on centralized security.

**enCFinterNetworkDeps** | **ci**
Select whether to carry forward internetwork dependencies. The default value is **yes**. Run `JnextPlan` to make this change effective.

**enCFResourceQuantity** | **rq**
Select whether resource quantities should be carried forward by stageman. The default value is **yes**. Run `JnextPlan` to make this change effective.

**enDbAudit** | **da**
Select whether to enable or disable database auditing. Valid values are **0** to disable database auditing, and **1** to activate database auditing. Auditing information is logged to a flat file in the `TWS_home/audit/database` directory. Each Tivoli Workload Scheduler workstation maintains its own log. For the database, only actions are logged in the auditing file, not the delta of the action. For more information on this feature, see [Enabling the time zone feature](#). The default value is **0**. Changes to this parameter are effective immediately.

**enEmptySchedsAreSucc** | **es**
Select the behavior of schedules without any jobs. Valid values are **YES** and **NO**. If set to **YES**, the schedules that contain no jobs are set to **SUCC** after their dependencies are resolved. If set to **NO**, the schedules are left in READY status. The default value is **NO**. Run `JnextPlan` to make this change effective.

**enEventDrivenWorkloadAutomation** | **ed**
Used in event rule management. Enables or disables the event-driven workload automation feature. Enter **yes** to enable, **no** to disable. The default value is **yes**.

After disabling, you must run `JnextPlan` or stop the event processing server (with the `conman stopevtp` command).

After enabling, you must run `JnextPlan` and start the event processing server (with the `conman startevtp` command).

**enEventProcessorHttpsProtocol** | **eh**
Used in event rule management. Enables or disables the use of the HTTPS protocol to connect to the event processor server. Enter **yes** to enable, **no** to disable. The default value is **yes**. Run `JnextPlan` to make this change effective.

**enLegacyId** | **li**
Determine how job streams are to be named when operating in mixed environments with older versions of Tivoli Workload Scheduler managed by a version 8.4 master domain manager. Use
Setting global options with the optman command

this option to keep consistency in identifying the job streams in the plan. The value assigned to this option is read either when the production plan is created or extended or when submitting job streams in production using conman.

Valid values are YES and NO. When the plan is created or extended, if this option is set to NO, the job stream instance is assigned a new id following the normal mechanism of Tivoli Workload Scheduler version 8.4. In the Symphony file the job stream name is equal to this id. If the option is set to YES, the job stream instance is assigned an id (symphony id) equal to the job stream name. In the Symphony file the job stream name is equal to the real job stream name. If more instances of the same job stream are present, for every instance an id is generated with an alias that starts from the job stream name.

The default value is NO. Run JnextPlan to make this change effective.

enLegacyStartOfDayEvaluation | le
Specify how the start0fDay option is to be managed across the Tivoli Workload Scheduler network. This option requires that the enTimeZone option be set to YES to become operational. Valid values are YES and NO. If you set this option to YES, the value assigned to the start0fDay option on the master domain manager is converted to the local time zone set on each workstation across the network. If you set this option to NO, the value assigned to the start0fDay option on the master domain manager is not converted to the local time zone set on each workstation across the network.

The default value is NO. Run JnextPlan to make this change effective.

enListSecChk | sc
Enable list security check is an option that controls which objects in the plan the user is permitted to list when running a Job Scheduling Console query or a conman show command. If set to yes, objects in the plan returned from a query or show command are shown to the user only if the user has been granted the list permission in the security file. For the database this option takes immediate effect. For the plan, JnextPlan must be run for this option to take effect. The default value is yes. Run JnextPlan to make this change effective.

Note: Setting this option to yes affects how the Job Scheduling Console functions for the users defined in the security file.

enLogonBatch | lb
Automatically grant logon as batch. This is for Windows jobs only. If set to yes, the logon users for Windows jobs are automatically granted the right to Logon as batch job. If set to no, or omitted, the right must be granted manually to each user or group. The right cannot be granted automatically for users running jobs on a Backup Domain Controller (BDC), so you must grant those rights manually. The default value is no. Run JnextPlan to make this change effective.

enPlanAudit | pa
Select whether to enable plan auditing. Valid values are 0 to
Setting global options with the optman command

disable plan auditing, and 1 to activate plan auditing. Auditing information is logged to a flat file in the TWShome/audit/plan directory. Each Tivoli Workload Scheduler workstation maintains its own log. For the plan, only actions are logged in the auditing file, not the success or failure of any action. The default value is 1. Run JnextPlan to make this change effective.

enPreventStart | ps
Select whether to prevent job streams without an at dependency from starting immediately, without waiting for the run cycle specified in the job stream. Valid values are yes and no. The default value is yes. Run JnextPlan to make this change effective.

enRetainNameOnRerunFrom | rr
Retain rerun job name is a production option that affects the operation of Batchman, which is the production control process of Tivoli Workload Scheduler. Its setting determines whether or not jobs that are rerun with the Conman rerun command will retain their original job names. Enter yes to have rerun jobs retain their original job names. The default values is no. Run JnextPlan to make this change effective.

enStrEncrypt | se
Select whether to enable strong encryption. The default value is no. See “Configuring the SSL connection protocol for the network” on page 170. Run JnextPlan to make this change effective.

enSwfaultTol | Sw
Select whether to enable or disable the fault tolerant switch manager. Valid values are yes and no. The default value is no. See “Switch fault tolerance” on page 9. Run JnextPlan to make this change effective.

enTimeZone | tz
Select whether to enable or disable the time zone option. Valid values are yes to activate time zones in your network and no to disable time zones in the network. For the database this option takes effect immediately. The default value is yes. Run JnextPlan to make this change effective. See Enabling the time zone feature

eventProcessorEIFPort | ee
Used in event rule management. Enter the port number where the event processor server receives events via the Tivoli Event Integration Facility (EIF). Valid values are in the 0-65535 range. Restart the Websphere Application Server and run JnextPlan to make this change effective.

extRecPrompt | xp
Specify an additional number of prompts for the value defined in baseRecPropmt for the case when a job is rerun after abending and the limit specified in baseRecPropmt has been reached. The default value is 1000. Run JnextPlan to make this change effective.

ignoreCals | ic
Ignore calendars is a preproduction option that affects the operation of the compiler command. Its setting determines whether or not user calendars are copied into the new Production Control file. Enter yes to prevent user calendars from being copied
Setting global options with the optman command

into the new production plan (Symphony file). The default value is no. See Tivoli Workload Scheduler Reference Guide. Run JnextPlan to make this change effective.

logCleanupFrequency | lc
Used in event rule management. Specify how often you want the automatic cleanup of log instances to be run. Valid values are in the 0-60 minutes range. If you specify 0, the automatic cleanup feature is disabled. The default value is 5 minutes. Run JnextPlan to make this change effective.

logHistory | lh
Used in event rule management. Enter the number of days for which you want to save rule instance, action run, and message log data. Log instances are discarded on a FIFO (first-in first-out) basis. The default value is 10 days. Run JnextPlan to make this change effective.

logmanMinMaxPolicy | lm
Select how the minimum and maximum job run times are logged and reported. Possible values are:

both 
Both the minimum and maximum job runtimes are logged and reported.

elapsedtime 
The minimum and maximum elapsed runtimes are logged and reported.

cputime 
The minimum and maximum cpu runtimes are logged and reported.

The default value is both. Run JnextPlan to make this change effective.

logmanSmoothPolicy | lt
Set the weighting factor that favors the most recent job run when calculating the normal (average) run time for a job. This is expressed as a percentage. For example, -smooth 40 applies a weighting factor of 40% to the most recent job run, and 60% to the existing average. The default value is -1. Run JnextPlan to make this change effective.

longDurationThreshold | ld
Specifies, when comparing the actual duration of a job to the planned duration, the threshold over which the job is considered to be of "long duration". The threshold value is expressed as a percentage with respect to the planned duration. For example, if the threshold is set to 150, and an actual duration is more than 150% of the planned duration (it is 50% greater), the job is considered to be a "long duration" job. Valid values are between:

- 100 (minimum) meaning that all jobs that exceed the planned duration are considered long duration jobs
- 1000 (maximum) meaning that only those jobs that last ten times as long as their planned duration are considered as long duration jobs

The default value is 150. Run JnextPlan to make this change effective.

mailSenderName | ms
Used in event rule management. If you deploy rules implementing
Setting global options with the optman command

an action that sends e-mails via an SMTP server, enter the name of
who sends the e-mails. Changes to this parameter will be effective
immediately.

maxLen | xl The maximum length of the preproduction plan in days. The
default is 14 days. maxLen must be greater than or equal to
minLen and must be in the range of 14 to 365. Run jnextPlan to
make this change effective.

minLen | ml The minimum length of the preproduction plan in days. The
default is 8 days. Run jnextPlan to make this change effective.

smtpServerName | sn Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, write here the
name of the SMTP server. Changes to this parameter will be
effective immediately.

smtpServerPort | sp Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, write here the
port number used to connect to the SMTP server. Changes to this
parameter will be effective immediately.

smtpUseAuthentication | ua Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, specify here if the
SMTP connection needs to be authenticated. Values are yes or no.
The default is no. Changes to this parameter will be effective
immediately.

smtpUserName | un Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, write here the
SMTP user name. Changes to this parameter will be effective
immediately.

smtpUserPassword | up Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, write here the
SMTP user password. The password will be stored in an encrypted
form. Changes to this parameter will be effective immediately.

smtpUseSSL | us Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, specify here if the
SMTP connection is to be authenticated via SSL. Values are yes or
no. The default is no. Changes to this parameter will be effective
immediately.

smtpUseTLS | tl Used in event rule management. If you deploy rules implementing
an action that sends e-mails via an SMTP server, specify here if the
SMTP connection is to be authenticated via the Transport Layer
Security (TLS) protocol. Values are yes or no. The default is no.
Changes to this parameter will be effective immediately.

startOfDay | sd Enter the start time of the Tivoli Workload Scheduler processing
day in 24 hour format: hhmm (0000-2359). The default start time is
6:00 a.m., and the default launch time of the final job stream is
Setting global options with the optman command

5:59 a.m. If you change this option, you must also change the launch time of the final job stream, which is usually set to one minute before the start time. The default value is 0600. Run JnextPlan to make this change effective.

statsHistory | sh
Enter the number of days for which you want to save job statistics. Statistics are discarded on a first-in, first-out basis. This has no effect on job standard list files, which must be removed with the rmstldlist command. For the database this option takes effect immediately, however for the next plan period. See the Tivoli Workload Scheduler Reference Guide for information about the rmstldlist command. The default values is 10. Run JnextPlan to make this change effective.

TECServerName | th
Used in event rule management. If you use rules implementing an action that forwards events to a Tivoli Enterprise Console (TEC) server (or any other application that can process events in TEC format), enter the TEC server name. You can change this value when you define the action if you want to use a different TEC server. Restart the Websphere Application Server to make this change effective.

TECServerPort | tp
Used in event rule management. If you use rules implementing an action that forwards events to a Tivoli Enterprise Console (TEC) server (or any other application that can process events in TEC format), enter the port number of the TEC server. The default port number is 5529. You can change this value when you define the action if you want to use a different TEC server. Restart the Websphere Application Server to make this change effective.

Setting local options

You set local options in the localopts file. Changes do not take effect until Tivoli Workload Scheduler is stopped and restarted.

A template file containing default settings is located in TWS/home/config/localopts.

Note: All of the SSL settings in the localopts file relate to the network communications and do not relate to the Job Scheduling Console.

During the installation process, a working copy of the local options file is installed as TWS/home/localopts.

localopts consists of the following syntax:

```
# comment
appserver auto restart = on|off
appserver check interval = minutes
appserver count reset interval = hours
appserver min restart time = minutes
appserver max restarts = number
appserver service name = name
autostart monman = yes|no
bm check deadline = seconds
bm check file = seconds
bm check status = seconds
bm check until = seconds
```
Setting local options in the `localopts` file

```
bm look = seconds
bm read = seconds
bm stats = on|off
bm verbose = on|off
can be event processor = yes|no
clisslcipher = string
clisslserverauth = yes|no
clisslservercertificate = filename
clissltrusteddir = directory_name
composer prompt = key
conman prompt = key
date format = integer
defaultws = master_workstation
host = host_name
is remote cli = yes|no
jm job table size = entries
jm look = seconds
jm nice = value
jm no root = yes|no
jm read = seconds
local was = yes|no
merge logtrace = yes|no
merge stdlists = yes|no
mm cache mailbox = yes|no
mm cache size = bytes
mm resolve master = yes|no
mm response = seconds
mm retrylink = seconds
mm sound off = yes|no
mm start tomserver = yes|no
mm unlink = seconds
mozaic directory = mozart_share
mm ipvalidate = none|full
mm mortal = yes|no
mm port = port number
mm read = seconds
mm retry = seconds
mm SSL port = value
parameters directory = parms_share
port = port number
protocol = protocol
proxy = proxy server
proxyport = proxy server port number
SSL auth mode = caonly|string|cpu
SSL auth string = string
SSL CA certificate = *.crt
SSL certificate = *.crt
SSL certificate chain = *.crt
SSL encryption cipher = shortcut
SSL key = *.key
SSL key pwd = *.sth
SSL random seed = *.rnd
stdlist width = columns
sync level = low|medium|high
switch sym prompt = key
syslog local = facility
tcp timeout = seconds
thiscpu = workstation
timeout = timeout
unison network directory = unison_share
useropts = useropts_file
wr read = seconds
wr enable compression = yes|no
wr unlink = seconds
```

Note: `localopts` syntax is not case-sensitive.
Setting local options in the \texttt{localopts} file

where:

\begin{itemize}
  \item \texttt{# comment} \quad Treats everything from the pound sign to the end of the line as a comment.
  \item \texttt{appserver auto restart} \quad Requests the \texttt{appserverman} process to automatically start WebSphere Application Server if it is found down. The default is \texttt{ON}.
  \item \texttt{appserver check interval} \quad Specifies the frequency in minutes that the \texttt{appserverman} process is to check that WebSphere Application Server is still running. The default is 5 minutes.
  \item \texttt{appserver count reset interval} \quad Specifies the time interval in hours after which the restart count is reset from the last WebSphere Application Server start. The default is 24 hours.
  \item \texttt{appserver max restarts} \quad Specifies the maximum number of restarting attempts the \texttt{appserverman} process can make before giving up and exiting without restarting WebSphere Application Server. The counter is reset if WebSphere Application Server runs for longer than the \texttt{appserver count reset interval} value. The default is 5.
  \item \texttt{appserver min restart time} \quad Specifies in minutes the minimum elapsed time the \texttt{appserverman} process must wait between each attempt to restart the WebSphere Application Server if it is down. If this value is less than the \texttt{appserver check interval}, the WebSphere Application Server will be restarted as soon as it is found down. If it is found down before this time (min restart time) interval has elapsed, \texttt{appserverman} will exit without restarting it. The default is 10 minutes.
  \item \texttt{appserver service name} \quad Only in Windows environments. Specifies the name of the WebSphere Application Server windows service if different from the standard name. This field is generally not used.
  \item \texttt{autostart monman} \quad Used in event rule management. Restarts the monitoring engine automatically when the next production plan is activated (on Windows also when IBM Tivoli Workload Scheduler is restarted). The default is \texttt{Yes}.
  \item \texttt{bm check deadline} \quad Specify the frequency in seconds that Batchman checks if any job or job stream instances have exceeded their deadline time. This option applies to jobs or job streams defined with the \texttt{deadline} keyword (see the \textit{Planning and Installation} guide for reference). The default is 0.
  \item \texttt{bm check file} \quad Specify the minimum number of seconds Batchman waits before checking for the existence of a file that is used as a dependency. The default is 120 seconds.
  \item \texttt{bm check status} \quad Specify the number of seconds Batchman waits between checking the status of an internetwork dependency. The default is 300 seconds.
\end{itemize}
Setting local options in the localopts file

**bm check until**
Specify the maximum number of seconds Batchman waits before reporting the expiration of an Until time for job or job stream. Specifying a value below the default setting (300) might overload the system. If it is set below the value of Local Option **bm read**, the value of **bm read** is used in its place. The default is 300 seconds.

**bm look**
Specify the minimum number of seconds Batchman waits before scanning and updating its production control file. The default is 15 seconds.

**bm read**
Specify the maximum number of seconds Batchman waits for a message in the INTERCOM.MSG message file. If no messages are in queue, Batchman waits until the timeout expires or until a message is written to the file. The default is 10 seconds.

**bm stats**
Specify on to have Batchman send its startup and shutdown statistics to its standard list file. Specify off to prevent Batchman statistics from being sent to its standard list file. The default is off.

**bm verbose**
Specify on to have Batchman send all job status messages to its standard list file. Specify off to prevent the extended set of job status messages from being sent to the standard list file. The default is off.

**can be event processor**
Specify if this workstation can act as event processing server or not. It is set by default to yes for master domain managers and backup masters, otherwise it is set to no.

**clisslcipher**
Specify the string that is used for the SSL authentication when the command-line client and the server are using SSL authentication.

**clisslserverauth**
Specify yes if the command-line client and the server use SSL authentication in their communication. The default is no.

**clisslservercertificate**
Specify the file that contains the SSL certificate when the command-line client and the server use SSL authentication in their communication.

**clissltrusteddir**
Specify the directory that contains an SSL trusted certificate contained in files with hash naming (#) when the command-line client and the server are using SSL authentication in their communication. When the directory path contains blanks enclose it in double-quotes.

**composer prompt**
Specify a prompt for the composer command line. The prompt can be of up to 10 characters in length. The default is dash (-).

**conman prompt**
Specify a prompt for the conman command line. The prompt can be of up to 8 characters in length. The default is percent (%).

**date format**
Specify the value that corresponds to the date format you desire. The values can be:
- 0 corresponds to yy/mm/dd
Setting local options in the **localopts file**

- 1 corresponds to **mm/dd/yy**
- 2 corresponds to **dd/mm/yy**
- 3 indicates usage of Native Language Support variables

The default is 1.

**defaultws**  The default workstation when you are accessing using a remote command-line. Specify the domain manager workstation.

**host**  The name or IP address of the host when accessing using a remote command-line.

**is remote cli**  Specify if this instance is installed as a remote command line client.

**jm job table size**  Specify the size, in number of entries, of the job table used by Jobman. The default is 1024 entries.

**jm look**  Specify the minimum number of seconds Jobman waits before looking for completed jobs and performing general job management tasks. The default is 300 seconds.

**jm nice**  For UNIX and Linux operating systems only, specify the **nice** value to be applied to jobs launched by Jobman. The default is 0.

**jm no root**  For UNIX and Linux operating systems only, specify **yes** to prevent Jobman from launching **root** jobs. Specify **no** to allow Jobman to launch **root** jobs. The default is **no**.

**jm read**  Specify the maximum number of seconds Jobman waits for a message in the COURIER.MSG message file. The default is 10 seconds.

**local was**  For master domain managers and backup masters connected to the Tivoli Workload Scheduler database. If set to **yes**, it improves the performance of job stream and job submission from the database. The default is **no**.

**merge logtrace**  Specify **yes** to merge the log and trace messages into a single log file. The default is **no**.

**merge stdlists**  Specify **yes** to have all of the Tivoli Workload Scheduler control processes, except Netman, send their console messages to a single standard list file. The file is given the name **TWSmerge**. Specify **no** to have the processes send messages to separate standard list files. The default is **yes**.

**mm cache mailbox**  Use this option to enable mailman to use a reading cache for incoming messages. In that case not all messages are cached, but only those not considered essential for network consistency. The default is **no**.

**mm cache size**  Specify this option if you also use **mm cache mailbox**. The default is 32. Use the default value for small and medium networks. Use larger values for large networks. The maximum value is 512.

**mm resolve master**  When set to **yes** the $MASTER variable is resolved at the beginning of the production day. The host of any extended agent is switched after the next **JnextPlan** (long term switch). When it is set
Setting local options in the `localopts` file

to **no**, the `$MASTER` variable is not resolved at **JnextPlan** and the host of any extended agent can be switched after a conman **switchmgr** command (short- and long-term switch). The default is **yes**. When you switch a master domain manager and the original has **mm resolve master** set to no and the backup has **mm resolve master** set to yes, after the switch any extended agent that is hosted by `$MASTER` switches to the backup domain manager. When the backup domain manager restarts, the keyword `$MASTER` is locally expanded by mailman. You should keep the **mm resolve master** value the same for master domain managers and backup domain managers.

**mm response** Specify the maximum number of seconds Mailman waits for a response before reporting that a workstation is not responding. The minimum wait time for a response is 90 seconds. The default is 600 seconds.

**mm retrylink** Specify the maximum number of seconds Mailman waits after unlinking from a non-responding workstation before it attempts to link to the workstation again. The default is 600 seconds. The **tomserver** optional mailman servers do not unlink non-responding agents. The link is repetitively checked every 60 seconds, which is the default **retrylink** for these servers.

**mm sound off** Specifies how Mailman responds to a conman **telop** ? command. Specify **yes** to have Mailman display information about every task it is performing. Specify **no** to have Mailman send only its own status. The default is **no**.

**mm start tomserver**

Specify **yes** on the domain manager to define a local mailman server named **tomserver** (also referred to as **server@** in the log files) that takes charge of the process of forwarding all the events meant for the parent domain manager. When there is no link to the parent domain manager, the events are stored in the **tomaster** pobox of the domain manager. As soon as the link is reestablished, **tomserver** transfers all the messages stored in **tomaster** to the parent domain manager. The advantage of using this server to free mailman of the task of upstream messaging (inclusive of checking the link with the parent domain manager) is twofold:

- The transmission of the Symphony file down to the network is faster
- Mailman can dedicate all its resources to other tasks.

The **tomserver** is incompatible with the fault tolerant switch mechanism. The default is **no**.

**mm unlink** Specify the maximum number of seconds Mailman waits before unlinking from a workstation that is not responding. The wait time should not be less than the response time specified for the Local Option **mm response**. The default is 960 seconds.

**mozart directory**

Defines the name of the master’s shared mozart directory. The default is none.

**mm ipvalidate** Specify **full** to enable IP address validation. If IP validation fails, the connection is not allowed. Specify **none** to allow connections when IP validation fails. The default is **none**.
Setting local options in the localopts file

**nm mortal**  Specify yes to have Netman quit when all of its child processes have stopped. Specify no to have Netman keep running even after its child processes have stopped. The default is no.

**nm port**  Specify the TCP port number that Netman responds to on the local computer. This must match the TCP/IP port in the computer’s workstation definition. It must be an unsigned 16-bit value in the range 1-65535 (values between 0 and 1023 are reserved for services such as, FTP, TELNET, HTTP, and so on). The default is 31111.

**nm read**  Specify the maximum number of seconds Netman waits for a connection request before checking its message queue for stop and start commands. The default is 10 seconds.

**nm retry**  Specify the maximum number of seconds Netman waits before retrying a connection that failed. The default is 800 seconds.

**nm SSL port**  The port used to listen for incoming SSL connections. This value must match the one defined in the secureaddr attribute in the workstation definition in the database. It must be different from the nm port local option that defines the port used for normal communication.

**Notes:**
1. If you install multiple instances of Tivoli Workload Scheduler on the same computer, set all SSL ports to different values.
2. If you plan not to use SSL, set the value to 0.

The default is none.

**parameters directory**  Defines the name of the master’s shared TWShome directory. The default is none.

**port**  The TCP/IP port number of the protocol used when accessing using a remote command-line. The default is 31115.

**protocol**  The protocol used to connect to the host when accessing using a remote command-line.

**proxy**  The name of the proxy server used when accessing using a remote command-line.

**proxyport**  The TCP/IP port number of the proxy server used when accessing using a remote command-line.

**SSL auth mode**  The behavior of Tivoli Workload Scheduler during an SSL handshake is based on the value of the SSL auth mode option as follows:

- **caonly**  Tivoli Workload Scheduler checks the validity of the certificate and verifies that the peer certificate has been issued by a recognized CA. Information contained in the certificate is not examined. **caonly** the default.

- **string**  Tivoli Workload Scheduler checks the validity of the certificate and verifies that the peer certificate has been issued by a recognized CA. It also verifies that the Common Name (CN) of the Certificate Subject matches the string specified into the SSL auth string option. See [157](#).

- **cpu**  Tivoli Workload Scheduler checks the validity of the
Setting local options in the localopts file

certificate and verifies that the peer certificate has been issued by a recognized CA. It also verifies that the Common Name (CN) of the Certificate Subject matches the name of the CPU that requested the service.

SSL auth string
Used in conjunction with the SSL auth mode option when the "string" value is specified. The SSL auth string (ranges from 1 — 64 characters) is used to verify the certificate validity. The default is "tws".

SSL CA certificate
In SSL authentication, the name of the file containing the trusted certification authority (CA) certificates required for authentication. The CAs in this file are also used to build the list of acceptable client CAs passed to the client when the server side of the connection requests a client certificate. This file is the concatenation, in order of preference, of the various PEM-encoded CA certificate files. The default is TwShome/ss1/TWSTrustedCA.crt. See “Configuring the SSL connection protocol for the network” on page 170.

SSL certificate
In SSL authentication, the name of the local certificate file. The default is TwShome/ss1/TWS.crt. See “Configuring the SSL connection protocol for the network” on page 170.

SSL certificate chain
In SSL authentication, the name of the file that contains the concatenation of the PEM-encoded certificates of certification authorities which form the certificate chain of the workstation’s certificate. This parameter is optional. The default is TwShome/ss1/TWScertificateChainCA.crt. See “Configuring the SSL connection protocol for the network” on page 170.

SSL encryption cipher
The ciphers that the workstation supports during an SSL connection. Use the following shortcuts:

Table 34. Shortcuts for encryption ciphers

<table>
<thead>
<tr>
<th>Shortcut</th>
<th>Encryption ciphers</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSLv3</td>
<td>SSL version 3.0</td>
</tr>
<tr>
<td>TLSv</td>
<td>TLS version 1.0</td>
</tr>
<tr>
<td>EXP</td>
<td>Export</td>
</tr>
<tr>
<td>EXPORT40</td>
<td>40-bit export</td>
</tr>
<tr>
<td>EXPORT56</td>
<td>56-bit export</td>
</tr>
<tr>
<td>LOW</td>
<td>Low strength (no export, single DES)</td>
</tr>
<tr>
<td>MEDIUM</td>
<td>Ciphers with 128 bit encryption</td>
</tr>
<tr>
<td>HIGH</td>
<td>Ciphers using Triple-DES</td>
</tr>
<tr>
<td>NULL</td>
<td>Ciphers using no encryption</td>
</tr>
</tbody>
</table>

The default is SSLv3.

SSL key
In SSL authentication, the name of the private key file. The default is TwShome/ss1/TWS.key. See “Configuring the SSL connection protocol for the network” on page 170.

SSL key pwd
In SSL authentication, the name of the file containing the password
Setting local options in the localopts file

for the stashed key. The default is TWShome/ssl/TWS.sth. See “Configuring the SSL connection protocol for the network” on page 120.

SSL random seed
The pseudo random number file used by OpenSSL on some operating systems. Without this file, SSL authentication might not work correctly. The default is TWShome/ssl/TWS.rnd. See “Configuring the SSL connection protocol for the network” on page 120.

stdlist width Specify the maximum width of the Tivoli Workload Scheduler console messages. You can specify a column number in the range 1 to 255 and lines are wrapped at or before the specified column, depending on the presence of imbedded carriage control characters. Specify a negative number or zero to ignore line width. On UNIX and Linux operating systems, you should ignore line width if you enable system logging with the syslog local option. The default is 0 columns.

syslog local Enables or disables Tivoli Workload Scheduler system logging for UNIX and Linux operating systems only. Specify -1 to turn off system logging for Tivoli Workload Scheduler. Specify a number from 0 to 7 to turn on system logging and have Tivoli Workload Scheduler use the corresponding local facility (LOCAL0 through LOCAL7) for its messages. Specify any other number to turn on system logging and have Tivoli Workload Scheduler use the USER facility for its messages. The default is -1. See “Tivoli Workload Scheduler console messages and prompts” on page 161.

sync level Specify the rate at which Tivoli Workload Scheduler synchronizes information written to disk. This option affects all mailbox agents and is applicable to UNIX and Linux operating systems only. Values can be:

low Allows the operating system to handle it.

medium Flushes the updates to disk after a transaction has completed.

high Flushes the updates to disk every time data is entered.

The default is low.

switch sym prompt Specify a prompt for the conman command line after you have selected a different Symphony file with the setsym command. The maximum length is 8 characters. The default is <n>.

tcp timeout With this attribute for the Netman process, specify the maximum number of seconds that Mailman and Conman waits for the completion of a request on a linked workstation that is not responding. The default is 300 seconds.

thiscpu Specify the Tivoli Workload Scheduler name of this workstation. When a switch is made between masters using the switchmgr command, the Symphony header value for thiscpu is overwritten by the thiscpu value in the localopts file. The default is thiscpu.
timeout The timeout in seconds when accessing using a remote command-line. The default is 3600 seconds.

unison network directory Defines the name of the Unison network directory. The default is none.

wr enable compression Use this option on fault-tolerant agents. Specify if the fault-tolerant agent can receive the Symphony file in compressed form from the master domain manager. The default is no.

wr read Specify the number of seconds the Writer process waits for an incoming message before checking for a termination request from Netman. The default is 600 seconds.

wr unlink Specify the number of seconds the Writer process waits before exiting if no incoming messages are received. The minimum is 120 seconds. The default is 180 seconds.

Local options file example

# The Tivoli Workload Scheduler localopts file defines the attributes of this workstation, for various processes.
#
# General attributes of this workstation:
#
thiscpu =MYFTA1
merge stdlists =yes
stdlist width =0
syslog local =-1
#
#
# The attributes of this workstation for the batchman process:
#
bm check file =120
bm check status =300
bm look =15
bm read =10
bm stats =off
bm verbose =off
bm check until =300
#
#
# The attributes of this workstation for the jobman process:
#
jm job table size =1024
jm look =300
jm nice =0
jm no root =no
jm read =10
#
#
# The attributes of this workstation for the TWS mailman process:
#
mm response =600
mm retrylink =600
mm sound off =no
mm unlink =960
mm cache mailbox =yes
mm cache size =512
mm resolve master =yes
Setting local options in the localopts file

```bash
autostart monman =yes
#
# The attributes of this workstation for the netman process:
#
nm mortal =no
nm port =31111
nm read =10
nm retry =800
#
#
# The attributes of this workstation for the writer process:
#
wrl read =600
wrl unlink =180
wrl enable compression =no
#
#
# Optional attributes of this workstation for remote database files
#
mozart directory = C:\TWS\tws84test/mozart
parameters directory = C:\TWS\tws84test
unison network directory = C:\TWS\tws84test/./unison/network
#
#
# The attributes of this workstation for custom formats
#
date format =1 # The possible values are 0-yyyy/mm/dd, 1-mm/dd/yyyy,
# 2-dd/mm/yyyy, 3-NLS.
composer prompt =-
conman prompt =
switch sym prompt =<n>
#
#
# The attributes of this workstation for the customization of I/O on mailbox files
#
sync level =low
#
#
# The attributes of this workstation for networking
#
tcp timeout =300
#
#
# The attributes of this workstation for SSL
#
nm SSL port =0
SSL key ="C:\TWS\tws84test/ssl/TWS.key"
SSL certificate ="C:\TWS\tws84test/ssl/TWS.crt"
SSL key pwd ="C:\TWS\tws84test/ssl/TWS.sth"
SSL certificate chain ="C:\TWS\tws84test/ssl/TWSCertificateChain.crt"
SSL random seed ="C:\TWS\tws84test/ssl/TWS.rnd"
SSL auth mode =caonly
SSL auth string =tws
#
#
# The attributes of this workstation for the embedded version of the
# WebSphere Application Server
#
LOCAL WAS =no
#
#
# Application server check attributes

Appserver check interval = 5 #minutes
Appserver auto restart = yes #yes/no
Appserver min restart time = 10 #minutes
```
Tivoli Workload Scheduler console messages and prompts

The Tivoli Workload Scheduler control processes (Netman, Mailman, Batchman, Jobman, and Writer) write their status messages, referred to as console messages, to standard list files. These messages include the prompts used as job and job stream dependencies. On UNIX and Linux operating systems, the messages can also be directed to the syslog daemon (syslogd) and to a terminal running the Tivoli Workload Scheduler console manager. These features are described in the following sections.

Setting sysloglocal on UNIX

If you set sysloglocal in the local options file to a positive number, Tivoli Workload Scheduler’s control processes send their console and prompt messages to the syslog daemon. Setting it to -1 turns this feature off. If you set it to a positive number to enable system logging, you must also set the local option stdlistwidth to 0, or a negative number.

Tivoli Workload Scheduler’s console messages correspond to the following syslog levels:

LOG_ERR Error messages such as control process abends and file system errors.

LOG_WARNING Warning messages such as link errors and stuck job streams.

LOG_NOTICE Special messages such as prompts and tellops.

LOG_INFO Informative messages such as job launches and job and job stream state changes.

Setting sysloglocal to a positive number defines the syslog facility used by Tivoli Workload Scheduler. For example, specifying 4 tells Tivoli Workload Scheduler to attach the useropts_tws84test file to the following system facility to log messages:

```
Useropts = useropts_tws84test
```

As an example of setting the logging parameters to log console messages to the syslog daemon, you can specify the following system facility in the following useropts file:
Tivoli Workload Scheduler console messages and prompt

use the local facility LOCAL4. After doing this, you must make the appropriate entries in the /etc/syslog.conf file, and reconfigure the syslog daemon. To use LOCAL4 and have the Tivoli Workload Scheduler messages sent to the system console, enter the following line in /etc/syslog.conf:

```
local4 /dev/console
```

To have the Tivoli Workload Scheduler error messages sent to the maestro and root users, enter the following command:

```
local4.err maestro,root
```

The selector and action fields must be separated by at least one tab. After modifying /etc/syslog.conf, you can configure the syslog daemon by entering the following command:

```
kill -HUP `cat /etc/syslog.pid`
```

console command

You can use the conman console command to set the Tivoli Workload Scheduler message level and to direct the messages to your terminal. The message level setting affects only Batchman and Mailman messages, which are the most numerous. It also sets the level of messages written to the standard list file or files and the syslog daemon. The following command, for example, sets the level of Batchman and Mailman messages to 2 and sends the messages to your computer:

```
console sess;level=2
```

Messages are sent to your computer until you either run another console command, or exit conman. To stop sending messages to your terminal, you can enter the following conman command:

```
console sys
```

Enabling the time zone feature

Time zones are enabled by default on installation of the product.

When you upgrade, the time zone feature it inherits the setting of the previous installation. You can enable the time zone using the enTimeZone option of the optman command, as follows:

```
optman chg enTimeZone = yes
```

The following steps outline the method of implementing the time zone feature:

1. Load Tivoli Workload Scheduler.
   The database allows time zones to be specified for workstations, but not on start and deadline times within job streams in the database. The plan creation (JnextPlan) ignores any time zones that are present in the database. You will not be able to specify any time zones anywhere in the plan.

2. Define workstation time zones.
   Set the time zone of the master workstation, of the backup master, and of any agents that are in a different time zone than the master. No time zones are allowed in the database for Start, Latest Start Time, and Termination Deadline times. No time zones are allowed anywhere in the plan at this point, because enTimeZone is set to NO.

3. When workstation time zones have been set correctly, enable the time zone feature.
Enabling the time zone feature

All users are able to use time zones anywhere in the database, although they should wait for the next run of JnextPlan to use them on Start, Latest Start Time, and Termination Deadline times. The next time JnextPlan runs, time zones are carried over to the plan and the Job Scheduling Console and the back end allows specification of time zones anywhere in the plan.

4. Start using time zones on start and until times where needed.
   You can now use all time zone references in the database and in the plan with both the Job Scheduling Console and the command-line interface.
Enabling the time zone feature
Chapter 11. Setting connection security

This chapter describes connection security. It is divided into the following sections:

- “Connection security overview”
- “Using an SSL connection for communication across the network”
- “Command-line client and Job Scheduling Console communication” on page 171
- “Configuring the Job Scheduling Console to work in a NAT environment” on page 176
- “Working across firewalls” on page 177
- “LDAP connection authentication” on page 178

Connection security overview

Tivoli Workload Scheduler uses the following protocols:

- SSL for communication across the network
- HTTP/HTTPS over SSL for the command-line client
- RMI/IIOP over SSL for Job Scheduling Console

The Tivoli Workload Scheduler Version 8.4 password encryption algorithm uses 3DES, which is also known as Triple-DES, or DES-FDE3. The algorithm is run in Cipher Block Chaining Mode, which uses padding according to the PKCS#5 standard.

The product is installed with default settings that you customize according to your security requirements.

Before you perform any customization of the SSL connections, stop the WebSphere Application server. Depending on the security settings you implement in your network, you will need to perform one or more tasks on the WebSphere Application Server. These tasks are described in Administration and Troubleshooting

Using an SSL connection for communication across the network

Tivoli Workload Scheduler provides a secure, authenticated, and encrypted connection mechanism for communication across the network topology. This mechanism is based on the Secure Sockets Layer (SSL) protocol and uses the OpenSSL Toolkit, which is automatically installed with Tivoli Workload Scheduler.

The SSL protocol is based on a private and public key methodology. SSL provides the following authentication methods:

CA trusting only

Two workstations trust each other if each receives from the other a certificate that is signed or is trusted. That is, if the CA certificate is in the list of trusted CAs on each workstation. With this authentication level, a workstation does not perform any additional checks on certificate content, such as the distinguished name. Any signed or trusted certificate can be used to establish an SSL session. See “Setting local options” on page 150 for a definition of the caonly option.
Connection security overview

Check if the distinguished name matches a defined string
Two workstations trust each other if, after receiving a trusted or signed certificate, each performs a further check by extracting the distinguished name from the certificate and comparing it with a string that was defined in its local options file. See “Setting local options” on page 150 for a definition of the string option.

Check if the distinguished name matches the workstation name
Two workstations trust each other if, after receiving a trusted or signed certificate, each performs a further check by extracting the distinguished name from the certificate and comparing it with the name of the workstation that sent the certificate. See “Setting local options” on page 150 for a definition of the cpu option.

To provide SSL security for a domain manager attached to z/OS in an end-to-end connection, configure the OS/390 Cryptographic Services System SSL in the Tivoli Workload Scheduler code that runs in the OS/390 USS UNIX shell in the IBM Tivoli Workload Scheduler for z/OS server address space. See the Tivoli Workload Scheduler z/OS documentation.

When configuring SSL you can:

Use the same certificate for the entire network
If the workstations are configured with CA trusting only, they accept connections with any other workstation that sends a signed or trusted certificate. To enforce the authentication you define a name or a list of names that must match the contents of the certificate distinguished name (DN) field in the localopts file of each workstation.

Use a certificate for each domain
Install private keys and signed certificates for each domain in the network. Then, configure each workstation to accept a connection only with partners that have a particular string of the certificate DN field in the localopts file of each workstation.

Use a certificate for each workstation
Install a different key and a signed certificate on each workstation and add a Trusted CA list containing the CA that signed the certificate. Then, configure each workstation to accept a connection only with partners that have their workstation name specified in the Symphony file recorded in the DN field of the certificate.

Setting up private keys and certificates
To use SSL authentication on a workstation, you need to create and install the following:

- The private key and the corresponding certificate that identify the workstation in an SSL session.
- The list of certificate authorities that can be trusted by the workstation.

Use the openssl command line utility to:

- Create a file containing pseudo random generated bytes (TWS.rnd). This file is needed on some operating systems for SSL to function correctly.
- Create a private key.
- Save the password you used to create the key into a file.
- Create a Certificate Signing Request.
• Send this Certificate Signing Request (CSR) to a Certifying Authority (CA) for signing, or:
  – Create your own Certificate Authority (CA)
  – Create a self-signed CA Certificate (X.509 structure) with the RSA key of your own CA
  – Use your own Certificate Authority (CA) to sign and create real certificates

These actions will produce the following files that you will install on the workstation(s):

• A private key file (for example, TWS.key). This file should be protected, so that it is not stolen to use the workstation’s identity. You should save it in a directory that allows read access to the TWS user of the workstation, such as TWS/home/ssl/TWS.key.

• The corresponding certificate file (for example, TWS.crt). You should save it in a directory that allows read access to the TWS user of the workstation, such as TWS/home/ssl/TWS.crt.

• A file containing a pseudo-random generated sequence of bytes. You can save it in any directory that allows read access to the TWS user of the workstation, such as TWS/home/ssl/TWS.rnd.

In addition, you should create the following:

• A file containing the password used to encrypt the private key. You should save it in a directory that allows read access to the TWS user of the workstation, such as TWS/home/ssl/TWS.sth.

• The certificate chain file. It contains the concatenation of the PEM-encoded certificates of certification authorities which form the certificate chain of the workstation’s certificate. This starts with the issuing CA certificate of the workstation’s certificate and can range up to the root CA certificate. Such a file is simply the concatenation of the various PEM-encoded CA certificate files, usually in certificate chain order.

• The trusted CAs file. It contains the trusted CA certificates to use during authentication. The CAs in this file are also used to build the list of acceptable client CAs passed to the client when the server side of the connection requests a client certificate. This file is simply the concatenation of the various PEM-encoded CA certificate files, in order of preference.

Creating Your Own Certification Authority

If you are going to use SSL authentication within your company’s boundaries and not for outside internet commerce, you might find it simpler to create your own certification authority (CA) to trust all your IBM Tivoli Workload Scheduler installations. To do so, follow the steps listed below.

Note: In the following steps, the names of the files created during the process TWS and TWSca are sample names. You can use your own names, but keep the same file extensions.

1. Choose a workstation as your CA root installation.
2. Type the following command from the SSL directory to initialize the pseudo random number generator, otherwise subsequent commands may not work.
   • On UNIX:
     $ openssl rand -out TWS.rnd -rand ./openssl 8192
   • On Windows:
     $ openssl rand -out TWS.rnd -rand ./openssl.exe 8192
Connection security overview

3. Type the following command to create the CA private key:
   
   $ openssl genrsa -out TWSca.key 1024

4. Type the following command to create a self-signed CA Certificate (X.509 structure):
   
   $ openssl req -new -x509 -days 365 -key TWSca.key -out TWSca.crt -config ./openssl.cnf

Now you have a certification authority that you can use to trust all of your installations. If you wish, you can create more than one CA.

Creating private keys and certificates

The following steps explain how to create one key and one certificate. You can decide whether to use one key and certificate pair for the entire network, one for each domain, or one for each workstation. The steps below assume that you will be creating a key and certificate pair for each workstation and thus the name of the output files created during the process has been generalized to workstationname.

On each workstation, perform the following steps to create a private key and a certificate:

1. Enter the following command from the SSL directory to initialize the pseudo random number generator, otherwise subsequent commands may not work.
   
   - On Windows operating systems:
     
     $ openssl rand -out workstationname.rnd -rand ./openssl.exe 8192
   
   - On UNIX and Linux operating systems:
     
     $ openssl rand -out workstationname.rnd -rand ./openssl 8192

2. Enter the following command to create the private key (this example shows triple-DES encryption):
   
   $ openssl genrsa -des3 -out workstationname.key 1024
   
   Then, save the password that was requested to encrypt the key in a file named workstationname.pwd.

   **Note:** Verify that file workstationname.pwd contains just the characters in the password. For instance, if you specified the word maestro as the password, your workstationname.pwd file should not contain any CR or LF characters at the end (it should be 7 bytes long).

3. Enter the following command to save your password, encoding it in base64 into the appropriate stash file:
   
   $ openssl base64 -in workstationname.pwd -out workstationname.sth
   
   You can then delete file workstationname.pwd.

4. Enter the following command to create a certificate signing request (CSR):
   
   $ openssl req -new -key workstationname.key -out workstationname.csr -config ./openssl.cnf
   
   Some values—such as company name, personal name, and more—will be requested at screen. For future compatibility, you may specify the workstation name as the distinguished name.

5. Send the workstationname.csr file to your CA in order to get the matching certificate for this private key.

   Using its private key (TWSca.key) and certificate (TWSca.crt), the CA will sign the CSR (workstationname.csr) and create a signed certificate (workstationname.crt) with the following command:
   
   $ openssl x509 -req -CA TWSca.crt -CAkey TWSca.key -days 365
   
   -in workstationname.csr -out workstationname.crt -CAcreateserial
6. Distribute to the workstation the new certificate `workstationname.crt` and the public CA certificate `TWSca.crt`.

The table below summarizes which of the files created during the process have to be set as values for the workstation’s local options.

Table 35. Files for Local Options

<table>
<thead>
<tr>
<th>Local option</th>
<th>File</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL key</td>
<td><code>workstationname.key</code></td>
</tr>
<tr>
<td>SSL certificate</td>
<td><code>workstationname.crt</code></td>
</tr>
<tr>
<td>SSL key pwd</td>
<td><code>workstationname.sth</code></td>
</tr>
<tr>
<td>SSL ca certificate</td>
<td><code>TWSca.crt</code></td>
</tr>
<tr>
<td>SSL random seed</td>
<td><code>workstationname.rnd</code></td>
</tr>
</tbody>
</table>

Configuring SSL attributes

Use the composer command line or the Job Scheduling Console to update the workstation definition in the database. See the *Tivoli Workload Scheduler Reference Guide* or the *Tivoli Workload Scheduler Job Scheduling Console User’s Guide* for further reference.

Configure the following attributes:

`secureaddr`

Defines the port used to listen for incoming SSL connections. This value must match the one defined in the `nm SSL port` local option of the workstation. It must be different from the `nm port` local option that defines the port used for normal communications. If `securitylevel` is specified but this attribute is missing, 31113 is used as the default value.

`securitylevel`

Specifies the type of SSL authentication for the workstation. It must have one of the following values:

- **enabled**
  
  The workstation uses SSL authentication only if its domain manager workstation or another fault-tolerant agent below it in the domain hierarchy requires it.

- **on**
  
  The workstation uses SSL authentication when it connects with its domain manager. The domain manager uses SSL authentication when it connects to its parent domain manager. The fault-tolerant agent refuses any incoming connection from its domain manager if it is not an SSL connection.

- **force**
  
  The workstation uses SSL authentication for all of its connections and accepts connections from both parent and subordinate domain managers. It will refuse any incoming connection if it is not an SSL connection.

If this attribute is omitted, the workstation is not configured for SSL connections. In this case, any value for `secureaddr` will be ignored. You should also set the `nm ssl port` local option to 0 to be sure that this port is not opened by netman. The following table describes the type of communication used for each type of `securitylevel` setting.

Table 36. Type of communication depending on the `securitylevel` value.

<table>
<thead>
<tr>
<th>Fault-tolerant agent (domain manager)</th>
<th>Domain manager (parent domain manager)</th>
<th>Connection type</th>
</tr>
</thead>
</table>
Connection security overview

Table 36. Type of communication depending on the securitylevel value. (continued)

<table>
<thead>
<tr>
<th>-</th>
<th>-</th>
<th>TCP/IP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>TCP/IP</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>On</td>
<td>No connection</td>
<td>No connection</td>
</tr>
<tr>
<td>Force</td>
<td>No connection</td>
<td>No connection</td>
</tr>
<tr>
<td>-</td>
<td>On</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>Enabled</td>
<td>On</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>On</td>
<td>SSL</td>
<td>SSL</td>
</tr>
<tr>
<td>Force</td>
<td>SSL</td>
<td>SSL</td>
</tr>
<tr>
<td>-</td>
<td>Enabled</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>Enabled</td>
<td>Enabled</td>
<td>TCP/IP</td>
</tr>
<tr>
<td>On</td>
<td>SSL</td>
<td>SSL</td>
</tr>
<tr>
<td>Force</td>
<td>SSL</td>
<td>SSL</td>
</tr>
<tr>
<td>-</td>
<td>Force</td>
<td>No connection</td>
</tr>
<tr>
<td>Enabled</td>
<td>Force</td>
<td>SSL</td>
</tr>
<tr>
<td>On</td>
<td>Force</td>
<td>SSL</td>
</tr>
<tr>
<td>Force</td>
<td>Force</td>
<td>SSL</td>
</tr>
</tbody>
</table>

The following example shows a workstation definition that includes the security attributes:

cpuname MYWIN
os WNT
node apollo
tcpaddr 30112
secureaddr 32222
for maestro
autolink off
fullstatus on
securitylevel on
end

Configuring the SSL connection protocol for the network

To configure SSL for your network, perform the following steps:

1. Create an SSL directory under the TWShome directory. By default, the path TWShome\ssl is registered in the localopts file. If you create a directory with a name different from ssl in the TWShome directory, then update the localopts file accordingly.

2. Copy openssl.cnf and openssl.exe to the SSL directory.

3. Create as many private keys, certificates, and Trusted CA lists as you plan to use in your network.

4. For each workstation that will use SSL authentication:
   - Update its definition in the Tivoli Workload Scheduler database with the SSL attributes.
   - Add the SSL local options in the localopts file.

Although you are not required to follow a particular sequence, these tasks must all be completed to activate SSL support.
Connection security overview

In Tivoli Workload Scheduler, SSL support is available for the fault-tolerant agents only (including the master and the domain managers), but not for the extended agents. If you want to use SSL authentication for a workstation that runs an extended agent, you must specify this parameter in the definition of the host workstation of the extended agent.

Command-line client and Job Scheduling Console communication

The command-line client and Job Scheduling Console use SSL encryption and server authentication to communicate. The command-line client communicates using HTTP/HTTPS over SSL and the Job Scheduling Console uses RMI/IIOP over SSL.

The Tivoli Workload Scheduler connector, command-line client, and the Job Scheduling Console use default certificates that are installed into default keystores. To set SSL, you customize the defaults to the required security level. The default certificates are not used for Job Scheduling Console client authentication that is obtained using a user ID and password. The default passwords associated with each of the default keystores is default.

The SSL security paradigm implemented in the Websphere Application Server Application Server requires two keystores to be present on the clients and the server: a key keystore containing the private key and a trust keystore containing the certificates of the trusted counterparts. The connector uses the same private key stored in its key keystore to authenticate itself as server for both the Job Scheduling Console and command-line client, so changing the connector private key means the trusted server certificates for both the Job Scheduling Console and the command-line client must also be updated.

The default keystores are named:
- **JSCDefaultKeyFile.jks** and **JSCDefaultTrustFile.jks** located in **Job Scheduling Console_install_dir\keys** for the Job Scheduling Console
- **TWSServerKeyFile.jks** and **TWSServerTrustFile.jks** located in **Tivoli Workload Scheduler_install_dir\appserver\profiles\twprofile\etc** for the connector installed with a master domain manager
- **TWSServerKeyFile.jks** and **TWSServerTrustFile.jks** located in **Tivoli Workload Scheduler_install_dir\appserver\profiles\twconnprofile\etc** for the connector installed with a domain manager or fault-tolerant agent

Customizing the connector configuration files for SSL

You configure connector specifying the following:
- Name and location of the keystore
- Keystore password
- Keystore certificates

The name and location of the keystore and the keystore password are contained in the following files:
- **soap.client.props**
- **sas.client.props**
- **sas.stdclient.properties**
- **sas.tools.properties**

Chapter 11. Setting connection security 171
Connection security overview

located in the TWSHome\appserver\profiles\twsprofile\properties directory, and in the security.xml file located in the TWSHome/appserver/profiles/config/cells/DefaultNode/ directory.

The following is an example of the security.xml that contains those settings:

```xml
<repertoire xmi:id="SSLConfig_1" alias="DefaultNode/DefaultSSLSettings">
  <setting xmi:id="SecureSocketLayer_1"
    >
    keyFileName="${USER_INSTALL_ROOT}/etc/TWSServerKeyFile.jks"
    keyFilePassword="{xor}Ozo5PiozKw==" keyFileFormat="JKS"
    trustFileName="${USER_INSTALL_ROOT}/etc/TWSServerTrustFile.jks"
    trustFilePassword="{xor}Ozo5PiozKw==" trustFileFormat="JKS"
    clientAuthentication="false"
    securityLevel="HIGH"
    enableCryptoHardwareSupport="false">
    <cryptoHardware xmi:id="CryptoHardwareToken_1"/>
    <properties xmi:id="Property_6" name="com.ibm.ssl.protocol" value="SSL"/>
    <properties xmi:id="Property_7" name="com.ibm.ssl.contextProvider">
      value="IBMJSSE2"/>
  </setting> </repertoire>

You modify the connector configuration files using the script changeSecurityProperties located in the TWSHome\wastool directory. For instructions on how to do this see Administration and Troubleshooting. The following is a sample of the input:

```bash
# Repeat the following section for each connector

# SSL Panel

# keyFileName=$USER_INSTALL_ROOT/etc/TWSServerKeyFile.jks
# keyFilePassword=
# keyFileFormat=JKS
# trustFileName=$USER_INSTALL_ROOT/etc/TWSServerTrustFile.jks
# trustFilePassword=
# trustFileFormat=JKS
```

Changing a server key

You change a server key when you want the Job Scheduling Console or command-line client to safely authenticate with their server. Changing a connector private key means that the trusted server certificates for both the Job Scheduling Console and command-line client must also be updated.

This section describes how to apply changes to the connector side. "Customizing SSL connection for the Job Scheduling Console" on page 173 and "Customizing the SSL connection for a command-line client" on page 174 describe configuration for the Job Scheduling Console and command-line client.

You can customize certificates and update server keystores using ikeyman that is provided with the embedded version of WebSphere Application Server - Express, or using the keytool utility provided with the Java runtime environment. For the connector ikeyman is located in the directory TWSHome/appserver/bin. The keytool utility is located in TWSHome/appserver/java/jre/bin.

You can change the server key with a new self signed certificate that you generate directly, or with a certificate signed by a Certificate Authority.

If you use a self signed certificate for the server, replace the server private key in the server Key keystore and the server public key in the Trust keystores for all the Job Scheduling Console and command-line clients that connect to it.
Changing a server key

If you use a certificate signed by a Certificate Authority, replace the server private key in the server Key keystore and make sure you have the certificate of the Certificate Authority in the Trust keystores for all the Job Scheduling Console and command-line clients that connect to it.

The following procedure is an example of how to create a new server key pair using the **keytool** utility:

1. Generate an RSA key pair for the connector in the keystore:
   ```
   keytool -genkey -alias tws84 -dname "CN=TWS84, OU=Test, O=IBM, C=US"
   -keystore TWSServerKeyFile.jks -storepass default -keypass default
   -validity 3000 -keyalg RSA
   ```

2. Export the certificate to PEM format and import it into the keystore:
   ```
   %> keytool -export -alias -frc -file server.crt -keystore TWSServerKeyFile.jks
   -storepass default
   ```

When creating self signed certificates, do not use the DSA algorithm as the Tivoli Workload Scheduler command-line utilities cannot use it to establish SSL connection to the server. Make sure that the keys have the same password as the keystore where they are contained. Make sure you update the Trust keystores for the Job Scheduling Console and command-line client.

**Customizing SSL connection for the Job Scheduling Console**

When you change the connector private key you need to update the Trust keystore of the Job Scheduling Console with the public key pair when you use self signed certificates, or make sure that it contains the certificate of the Certificate Authority.

When you configure the Job Scheduling Console to connect to different connectors it must have a certificate in its Trust keystore that enables trust for each connector.

You can customize certificates and update the Job Scheduling Console Trust keystore using **ikeyman** that is provided with the embedded version of WebSphere Application Server - Express, or using the **keytool** utility provided with the Java runtime environment. For the Job Scheduling Console **ikeyman** and the **keytool** utility are located in the directory `install_dir/tools/_jvm/jre/bin`.

The following is an example of how to import the connector certificate in PEM format:

```
%>keytool -import -alias tws84 -file server.crt -trustcacerts
   -noprompt -keystore JSTrustFile.jks -storepass default
``` 

When you are customizing the Job Scheduling Console settings, make sure the keys have the same password as the keystore where they are contained. The keystore password must correspond between the connector server and the Job Scheduling Console client. The keys in the Job Scheduling Console Trust and Key keystores must be paired with the keys in the connector Key and Trust keystores respectively.

If you changed the Job Scheduling Console keystore password, name, or location, modify the following Job Scheduling Console configuration files when customizing SSL:

**NTConsole.bat**

Specifies the keystores and connection configuration files for the SSL connections with Java properties:
Customizing SSL connection for the Job Scheduling Console

com.ibm.ssl.keyStore
For the key keystore name.

com.ibm.ssl.trustStore
For the trusted keystore name.

com.ibm.CORBA.ConfigURL
For the connection properties file name.

NTConsole.bat is located in the install_dir/bin/java/ directory. For UNIX and Linux operating systems the corresponding scripts assume the operating system name, such as LINUXconsole.sh, SUNconsole.sh and so on.

sas.client.props
Specifies the default properties for SSL connection and contains the keystore names, types, and encoded keystore passwords. The following is an example:

```
# # Keystores
# com.ibm.ssl.keyStorePassword={xor}Ozo5Piozعامل
com.ibm.ssl.contextProvider=IBMJSSE
com.ibm.ssl.keyStoreType=JKS
com.ibm.ssl.trustStorePassword={xor}Ozo5Piozعامل
com.ibm.ssl.protocol=
com.ibm.ssl.trustStoreType=JKS
```

Encrypt the password using the PropFilePasswordEncoder utility. See Administration and Troubleshooting for details on how to encrypt profile properties.

Customizing the SSL connection for a command-line client

Tivoli Workload Scheduler command-line clients connect to the connector through HTTP or HTTPS. The default connection type is HTTPS. If the command-line connects through a proxy, use the HTTP connection protocol as HTTPS is not supported for this type of configuration.

You configure connection protocol modifying the following options in the localopts file:

**PROXY** Specify the IP address or the server name for the proxy server.

**PROTOCOL** Specify the protocol type as either HTTP or HTTPS. HTTPS is the default.

**PORT** Specify the port required by the protocol you set in the PROTOCOL option. The defaults are 31115 for HTTP and 31116 for HTTPS.

**PROXYPORT** Specify the port required by the proxy server, when you specified PROXY as the server type.

Alternatively, you can specify the entries in the useropts file. When you specify the options in the useropts file, they overwrite the options specified in the localopts file. See IBM Tivoli Workload Scheduler Reference Guide for more information. See “Setting local options” on page 150 for an explanation of these local options.

The HTTPS connection protocol offers the following additional security features:

- Data encryption between the command-line utility and the connector
Customizing SSL connection for the Job Scheduling Console

- Optional server authentication by validating the server certificates

You can activate optional server authentication in one of the following ways:

- “Configuring SSL using the predefined certificate”
- “Configuring multiple SSL communication instances”
- “Using a customized certificate”

Configuring SSL using the predefined certificate
To customize the SSL connection for the command-line client using the predefined certificate, perform the following steps:
1. Stop the embedded version of WebSphere Application Server as described in Administration and Troubleshooting.
2. Extract the certificate from the TWSServerKeyFile.jks keystore:
   ```
   keytool -export -alias tws84 -rfc -file server.crt
   -keystore TWSServerKeyFile.jks -storepass default
   ```
3. Copy the .crt certificate to each workstation that has a command-line client installed.
4. Set the `clissslserverauth` and `clissslservercertificate` command-line client options in the localopts file. See “Setting local options” on page 150.
5. Start the embedded version of WebSphere Application Server as described in Administration and Troubleshooting.

Configuring multiple SSL communication instances
To customize the SSL connection for the command-line client for multiple connections to WebSphere Application Server, perform the following steps:
1. Stop WebSphere Application Server See Administration and Troubleshooting
2. Extract a certificate from TWSServerKeyFile.jks keystore for each instance.
   ```
   keytool -export -alias tws84 -rfc -file server.crt -keystore ServerKeyFile.jks -storepass default
   ```
3. Extract the hash number for each exported certificate:
   ```
   openssl x509 -hash -noout -in keyname
   ```
4. Rename each certificate file with the exported key.
5. Copy the renamed certificates to each workstation that has a command-line client installed.
6. Set the `clissslserverauth` and `clisssltrusteddir` command-line client options in the localopts file. See “Setting local options” on page 150.
7. Start the WebSphere Application Server. See Administration and Troubleshooting for instructions on how to perform this task.

Using a customized certificate
To customize the SSL certificate and keystore, perform the following steps:
1. Create a new RSA and extract the key for the server keystore
   `TWSServerKeyFile.jks`.
2. Follow the steps in “Customizing SSL connection for the Job Scheduling Console” on page 175.
3. Follow the steps in “Configuring SSL using the predefined certificate.”

Note: When you want to customize certificates for multiple instances, perform these steps for each instance.
Customizing SSL connection for the Job Scheduling Console

Command-line SSL connection settings

Tivoli Workload Scheduler V8.4 command-line utilities connect to the connector through HTTP or HTTPS connections. The default connection type is HTTPS, but you can configure the connection as HTTP modifying entries in the localopts file. See “Setting local options” on page 150.

The secure connection protocol (HTTPS) offers the following additional security:

- Data encryption between the command-line utility and the connector
- Optional server authentication by validating the server certificates

When configuring HTTP or HTTPS for the command-line utilities, you need to provide user credentials for connection to the connector. You can do this in one of the following ways:

Using the command-line

You can provide the user credentials using the command-line parameters -username and -password.

Using the useropts file

You can specify the USERNAME and PASSWORD parameters in the useropts file. For example:

```
# USERNAME and PASSWORD
USERNAME = tws84
PASSWORD = "ENCRYPTMEPLEASE"
```

The password must be enclosed in double-quotes. For a complete definition of the user options, refer to Tivoli Workload Scheduler: Reference Guide.

At run time

When the user credentials are not specified using command-line parameters, or in the useropts file, the command-line requests them and automatically adds them to the useropts file.

If the command-line connects through a proxy, use the HTTP connection protocol. The HTTPs protocol is not supported in this type of configuration. Customize the following options in the localopts file if you connect through a proxy:

PROXY Specify the IP address or the server name for the proxy server.

PROXYPORT Specify the port number used by the proxy server.

Configuring the Job Scheduling Console to work in a NAT environment

The Job Scheduling Console and the connector use the RMI/IIOP protocol to communicate. Some of the connections between the Job Scheduling Console and the connector use JNDI and ORB Services. You can use the showHostProperties and changeHostProperties utilities to manage the parameters used in this phase of communication. See Administration and Troubleshooting.

You can set the following parameters:

bootHost

Specify the IP address or host name that is returned to the Job Scheduling Console and used to establish connections to the JNDI server. This address must be visible from the workstation where the Job Scheduling Console is running. This value is set during installation. Change the value to the public IP address or host name when you are running in a NAT.
Configuring the Job Scheduling Console to work in a NAT environment

**orbPort**

The default value is 0. Set the **orbPort** to a controlled port when running in a NAT environment.

**csiServerAuthPort**

The default value is 0. Set the **csiServerAuthPort** to a controlled port when running in a NAT environment.

The **bootHost**, **orbPort**, and **csiServerAuthPort** must be opened on firewalls and forwarded to NATs.

---

**Working across firewalls**

In the design phase of a Tivoli Workload Scheduler network, the administrator must know where the firewalls are positioned in the network, which fault-tolerant agents and which domain managers belong to a particular firewall, and which are the entry points into the firewalls. When this has been clearly understood, the administrator should define the **behindfirewall** attribute for some of the workstation definitions in the Tivoli Workload Scheduler database. In particular, if a workstation definition is set with the **behindfirewall** attribute to ON, this means that there is a firewall between that workstation and the Tivoli Workload Scheduler master. In this case, the workstation-domain manager link is the only link allowed between the workstation and its domain manager.

All Tivoli Workload Scheduler workstations should be defined with the **behindfirewall** attribute if the link with the corresponding domain manager, or with any domain manager in the Tivoli Workload Scheduler hierarchy right up to the master, is across a firewall.

All Tivoli Workload Scheduler workstations whose links with the corresponding domain manager or with any domain manager in the Tivoli Workload Scheduler hierarchy right up to the master, is across a firewall, should be defined with the **behindfirewall** attribute.

When mapping an IBM Tivoli Workload Scheduler network over an existing firewall structure, it does not matter which fault-tolerant agents and which domain managers are on the secure side of the firewall and which ones are on the non secure side. Firewall boundaries should be the only concern. For example, whether the master is in a non secure zone and some of the domain managers are in secured zones, or vice versa, does not make any difference. The firewall structure must always be considered starting from the master and following the Tivoli Workload Scheduler hierarchy, marking all the workstations that have a firewall between them and their corresponding domain manager.

For all workstations with **behindfirewall** set to ON, the command **start** and **stop** commands on the workstation, and the **showjobs** commands are sent following the domain hierarchy, instead of making the master or the domain manager open a direct connection to the workstation. This makes a significant improvement in security.

This attribute works for multiple nested firewalls as well. For extended agents, you can specify that an extended agent workstation is behind a firewall by setting the **behindfirewall** attribute to ON, on the host workstation. The attribute is read-only in the plan; to change it in the plan, the administrator must update it in the database and then recreate the plan.
Working across firewalls

See the Tivoli Workload Scheduler Reference Guide for details on how to set this attribute.

**LDAP connection authentication**

Tivoli Workload Scheduler uses LDAP (Lightweight Directory Access Protocol) to authenticate users to post-installation tasks using Websphere in the following environments:

- Microsoft Active Directory
- IBM Tivoli Directory Server
- Sun ONE DS (formerly known as IPLanet)

LDAP authenticated users can run configuration tasks from the following interfaces:

- composer command interface
- Job Scheduling Console
- Command-line client
- Command-line as clients connected to the master domain manager using HTTP or HTTPS

See Administration and Troubleshooting for details on LDAP configuration using the Websphere Application Server.
Appendix A. Integration with other IBM Tivoli products

This appendix describes integration with other IBM products. It is divided into the following sections:

- “Integration with Tivoli Enterprise Data Warehouse”
- “Integration with Tivoli NetView”
- “Integration with Tivoli Enterprise Console” on page 195
- “Integration with Tivoli Monitoring” on page 217
- “Integration with Tivoli Storage Manager” on page 232
- “Integration with Tivoli License Compliance Manager” on page 235

Integration with Tivoli Enterprise Data Warehouse

When your environment contains many products and services that manage and monitor your IT enterprise, storing this data, generating reports, and analyzing the data becomes a complex task. Tivoli Enterprise Data Warehouse enables you to collect this data in one place, a central data warehouse and enables you to construct an end-to-end view of your enterprise.

Tivoli Workload Scheduler provides a Tivoli Enterprise Data Warehouse enablement pack to consolidate scheduling data in the Tivoli Enterprise Data Warehouse database. The documentation for the warehouse enablement pack is on Tivoli Workload Scheduler Integrations CD labeled TWS_OTHER, in the path TDW_enablement_pack\tdw_weps\aws\v8400\doc\itws_for_TDW.pdf.

Integration with Tivoli NetView

This section describes the integration of Tivoli Workload Scheduler on UNIX with NetView for AIX. For a description of Tivoli NetView, refer to the Tivoli NetView documentation.

Tivoli Workload Scheduler/NetView

Tivoli Workload Scheduler/NetView consists of manager and agent software. The manager runs on NetView management nodes and the agent runs on managed nodes. All the nodes must have Tivoli Workload Scheduler installed. The manager polls its agents periodically to obtain information about scheduler processing. If the information returned during a poll is different from that of the preceding poll, the color of the corresponding symbol changes to indicate a state change. After you take action to remedy condition, the state of the corresponding symbol returns to normal at the next poll.

Agents generate SNMP traps to inform the manager of synchronous events, such as job abends, stuck schedules, and restarted scheduler processes. Although polling and traps are functionally independent, the information that accompanies a trap can be correlated with symbol state changes. If, for example, a scheduled job abends, the symbol for the workstation changes color and a job abend trap is logged in the NetView event log. By scanning the log, you can isolate the problem and take the appropriate action.
The **muser** process runs commands issued by a NetView user, and updates the user’s map. An **muser** is started for each NetView user whose map has the Tivoli Workload Scheduler/NetView application activate.

**Installing the integration software**

Tivoli Workload Scheduler/NetView is delivered as part of Tivoli Workload Scheduler. Before installing Tivoli Workload Scheduler/NetView, make sure that Tivoli Workload Scheduler is correctly installed on the management node (server and clients) and on each managed node and make sure it is successfully scheduling and tracking jobs.

Use the customize script to install the integration software. The customize script has the following syntax:

```
```

where:

- **[-uname name]**
  
  Tivoli Workload Scheduler user name.

- **[-prev3]**
  
  Include this option if your version of NetView is prior to version 3.

- **[-noinst]**
  
  Do not overwrite existing NetView configuration files.

- **[-client]**
  
  For NetView version 6.x and later, include this option for management clients.

- **[-manager]**
  
  The host name of the management node. For NetView version 6.x and above, this is the host name of the NetView server. This is required for managed nodes and NetView clients. Do not use this option on the management node or NetView server.

**Installing on managed nodes and NetView clients**

The management node can also be a managed node. For the management node or NetView server, skip this step and perform step [Installing on the management node or NetView server](#).

1. Make certain that no Tivoli Workload Scheduler processes are running. If necessary, issue a conman **shutdown** command.

2. Log in as root.

3. For managed nodes, including those that are also NetView clients that are not used to manage Tivoli Workload Scheduler, run the **customise** script as follows:

   ```
   /bin/sh <TWShome>/OV/customize -manager host
   ```

   where:

   **host** is the host name of the management node.

4. For NetView clients that are used to manage Tivoli Workload Scheduler, run **customise** as follows:

   ```
   /bin/sh <TWShome>/OV/customize -client [-manager host]
   ```

   where:

   **host** is the host name of the management node.

5. Run **StartUp**:

   ```
   <TWShome>/StartUp
   ```
Installing on the management node or NetView server

1. Make certain that no Tivoli Workload Scheduler processes are running. If necessary, issue a conman shutdown command.
2. Log in as root.
3. Run the customize script as follows:
   ```
   /bin/sh <TWShome>/OV/customize
   ```
4. If you do not want the Tivoli Workload Scheduler/NetView agent to run on this node, edit <TWShome>/StartUp, and remove the run of magent.
5. If you want Tivoli Workload Scheduler to run on this node, run StartUp:
   ```
   <TWShome>/StartUp
   ```
6. Start the Tivoli Workload Scheduler/NetView daemon (mdemon) as follows:
   ```
   /usr/OV/bin/ovstart Unison_Maestro_Manager
   ```
   or, for NetView versions below 3, stop and start as follows:
   ```
   /usr/OV/bin/ovstop
   /usr/OV/bin/ovstart
   ```

Setting up Tivoli Workload Scheduler/NetView

To setup Tivoli Workload Scheduler/NetView, perform the following steps:

1. On each managed node, enter the host name of the management node in the user’s $HOME/.rhosts file.
2. Add a user definition to the scheduler security file. For more information about Tivoli Workload Scheduler security, refer to the IBM Tivoli Workload Scheduler Reference Guide.
3. On the management node, run NetView.
4. Select Describe Map from the File menu.
5. Select Maestro-Unison Software (c) from the Configurable Applications list, and click Configure For This Map.
6. Click True under Enable Maestro™ for this map.
7. Click Verify.
8. Click OK to close the Configuration dialog box. Click OK to close the Map Description dialog box.

If you want to use the MIB browser, perform the following steps:

1. Select Load/Unload MIBs:SNMP from the Options menu.
2. Click Load.
3. When Load MIB From File opens, type:
   ```
   /usr/OV/snmp_mibs/Maestro.mib
   ```
   in MIB File to Load. Click OK.
4. Click Close.

If the management node is not also a Tivoli Workload Scheduler managed node, or if you manage more than one Tivoli Workload Scheduler network, use the NetView object description function to identify the managed nodes where Tivoli Workload Scheduler/NetView agents are running. To identify the managed nodes, perform the following steps:

1. Move down the IP Internet tree to the IP segment submap showing all the nodes.
Integration with Tivoli NetView

2. Select a node where a Tivoli Workload Scheduler/NetView agent is running. Press Ctrl-O to open the Object Description dialog.
3. On the Object Description dialog, select General Attributes from the Object Attributes list, and click View/Modify Object Attributes.
4. On the Attributes for Object dialog, click True under the isUTMaestroAgent attribute.
5. Click OK to close the Attributes for Object dialog. Click OK to close the Object Description dialog.
6. Repeat steps for each node where a Tivoli Workload Scheduler/NetView agent is running.
7. Return to the Root submap.
8. Select Tivoli Workload Scheduler, from the Tools menu.
10. When the Unison Software(c) symbol appears, double-click it to open the Unison Software(c) submap displaying a symbol for each Tivoli Workload Scheduler network. Double-click a network symbol to open a Network submap.

On the master domain issue a conman start@ command to start Tivoli Workload Scheduler in the network. This can be done in NetView on the Tivoli Workload Scheduler Network submap as follows:
1. Select all of the nodes in the network.
2. From the Tools menu, select Tivoli Workload Scheduler.
3. Select Start.

Menu actions

To use Tivoli Workload Scheduler/NetView menu actions, select Tivoli Workload Scheduler from the Tools menu. These actions are also available from the object context menu by right clicking a symbol.

The menu actions are:

View
- Open a child submap for a Tivoli Workload Scheduler/NetView symbol. Choosing View after selecting a workstation symbol on the submap opens the monitored processes submap. Choosing View after selecting a workstation symbol on the IP node submap returns to the Tivoli Workload Scheduler network submap.

Master conman
- Run the conman command-line on the Tivoli Workload Scheduler master. Running on the master lets you run conman commands (except shutdown) for any workstation in the network. For information about conman commands, see IBM Tivoli Workload Scheduler Reference.

Acknowledge
- Acknowledge the status of selected symbols. When acknowledged, the status of a symbol returns to normal. It is not necessary to acknowledge critical or marginal states for a monitored process symbol, as it returns to normal when the monitored process itself is running. Acknowledge critical or marginal states for workstation symbols, otherwise they do not return to normal.

Conman
- Run the conman command-line on the selected Tivoli Workload Scheduler workstations. Running on a workstation other than the master, lets you run all conman commands on that workstation.
only. For information about conman commands, see IBM Tivoli Workload Scheduler Reference. For an extended agent, conman is run on its host.

**Start**

Issue a conman **start** command for the selected workstations. By default, the command for this action is:

```
remsh %H %P/bin/conman "start %c"
```

**Down (stop)**

Issue a conman **stop** command for the selected workstations. By default, the command for this action is:

```
remsh %H %P/bin/conman "stop %c"
```

**StartUp**

Run the Tivoli Workload Scheduler **StartUp** script on the selected workstations. By default, the command for this action is:

```
remsh %h %P/StartUp
```

For an extended agent, **conman** is run on its host.

**Rediscover**

Locate new agents and new Tivoli Workload Scheduler objects, and update all Tivoli Workload Scheduler/NetView sub-maps.

**Note:** Run Rediscover each time you change the Tivoli Workload Scheduler workstation configuration.

The substituted parameters in the command-lines are:

- **%c**: The Tivoli Workload Scheduler workstation name of a selected workstation symbol.
- **%D**: The current DISPLAY name.
- **%h**: The host name of a selected workstation symbol.
- **%H**: The host name of the Tivoli Workload Scheduler master.
- **%p**: The process name of a selected process symbol, or “MAESTRO” if it is not a process.
- **%P**: The maestro user’s home directory (usually `/usr/lib/maestro`).

**Events**

Events 1-53 indicate the status of critical processes that are monitored by the Tivoli Workload Scheduler/NetView agents, including the agents themselves (event 1). Events 101-25) indicate the status of the job scheduling activity.

The listed events can result in SNMP traps generated by the Tivoli Workload Scheduler/NetView agents. Whether or not traps are generated is controlled by options set in the configuration files of the agents. See "Tivoli Workload Scheduler/NetView configuration files" on page 185 for more information.

The **Additional Actions** column lists the actions available. The actions can be initiated by selecting **Additional Actions** from the **Options** menu, then selecting an action from the Additional Actions panel.

**Note:** You need the appropriate Tivoli Workload Scheduler security access to perform the chosen action. See Tivoli Workload Scheduler: Reference Guide.
## Table 37. Tivoli Workload Scheduler/NetView events

<table>
<thead>
<tr>
<th>Trap #</th>
<th>Name</th>
<th>Description</th>
<th>Additional Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 *</td>
<td>uTtrapReset</td>
<td>The magent process was restarted.</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>uTtrapProcessReset</td>
<td>A monitored process was restarted. This event is reported by default in the BmEvents.conf file</td>
<td></td>
</tr>
<tr>
<td>52 *</td>
<td>uTtrapProcessGone</td>
<td>A monitored process is no longer present.</td>
<td></td>
</tr>
<tr>
<td>53 *</td>
<td>uTrapProcessAbend</td>
<td>A monitored process abended.</td>
<td></td>
</tr>
<tr>
<td>54 *</td>
<td>uTrapXagentConnLost</td>
<td>The connection between a host and xagent has been lost.</td>
<td></td>
</tr>
<tr>
<td>101 *</td>
<td>uTtrapJobAbend</td>
<td>A scheduled job abended.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>102 *</td>
<td>uTtrapJobFailed</td>
<td>An external job is in the error state.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>103</td>
<td>uTtrapJobLaunch</td>
<td>A scheduled job was launched successfully.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>104</td>
<td>uTtrapJobDone</td>
<td>A scheduled job finished in a state other than ABEND.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>105*</td>
<td>uTtrapJobUntil</td>
<td>A scheduled job’s UNTIL time has passed, it will not be launched.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>111</td>
<td>uTrapJobCant</td>
<td>A scheduled job could not be launched.</td>
<td>Show Job, Rerun Job, Cancel Job</td>
</tr>
<tr>
<td>151 *</td>
<td>uTtrapSchedAbend</td>
<td>A schedule ABENDed.</td>
<td>Show Schedule, Cancel Schedule</td>
</tr>
<tr>
<td>152 *</td>
<td>uTtrapSchedStuck</td>
<td>A schedule is in the STUCK state.</td>
<td>Show Schedule, Cancel Schedule</td>
</tr>
<tr>
<td>153</td>
<td>uTtrapSchedStart</td>
<td>A schedule has started execution.</td>
<td>Show Schedule, Cancel Schedule</td>
</tr>
<tr>
<td>154</td>
<td>uTtrapSchedDone</td>
<td>A schedule has finished in a state other than ABEND.</td>
<td>Show Schedule, Cancel Schedule</td>
</tr>
<tr>
<td>155*</td>
<td>uTtrapSchedUntil</td>
<td>A schedule’s UNTIL time has passed, it will not be launched.</td>
<td>Show Schedule, Cancel Schedule</td>
</tr>
<tr>
<td>201 *</td>
<td>uTtrapGlobalPrompt</td>
<td>A global prompt has been issued.</td>
<td>Reply</td>
</tr>
<tr>
<td>202 *</td>
<td>uTtrapSchedPrompt</td>
<td>A schedule prompt has been issued.</td>
<td>Reply</td>
</tr>
<tr>
<td>203 *</td>
<td>uTtrapJobPrompt</td>
<td>A job prompt has been issued.</td>
<td>Reply</td>
</tr>
<tr>
<td>204 *</td>
<td>uTtrapJobRerunPrompt</td>
<td>A job rerun prompt has been issued.</td>
<td>Reply</td>
</tr>
</tbody>
</table>
Table 37. Tivoli Workload Scheduler/NetView events (continued)

<table>
<thead>
<tr>
<th>Trap #</th>
<th>Name</th>
<th>Description</th>
<th>Additional Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>251</td>
<td>uTtrapLinkDropped</td>
<td>The link to a workstation has closed.</td>
<td>Link</td>
</tr>
<tr>
<td>252 *</td>
<td>uTtrapLinkBroken</td>
<td>The link to a workstation has closed due to an error.</td>
<td>Link</td>
</tr>
</tbody>
</table>

* These traps are enabled by default.

Polling and SNMP traps
Because SNMP uses an unreliable transport protocol (UDP), Tivoli Workload Scheduler/NetView does not rely on SNMP traps to indicate the status of its symbols. The manager polls its agents periodically, requesting specific MIB values. The returned values are compared with those returned by the previous poll, and differences are indicated as status changes in Tivoli Workload Scheduler/NetView symbols. The default polling interval is one minute. See “Tivoli Workload Scheduler/NetView configuration options” on page 189 for information about changing the polling interval.

To obtain critical process status, the manager polls all of its agents. For job scheduling status, the manager determines which of its agents is most likely to have the required information, and polls only that agent. The choice is made in the following order of precedence:
1. The agent running on the Tivoli Workload Scheduler master
2. The agent running on a Tivoli Workload Scheduler backup master
3. The agent running on any Tivoli Workload Scheduler fault-tolerant agent that has full status on in its workstation definition

Enabling traps provides the following advantages:
1. Event-specific variables are included with each trap
2. Traps are logged in NetView’s event log.

If job abend traps (101) are enabled, for example, sufficient information is collected to identify an abended job, its schedule, and the workstation on which it runs. This is useful when deciding what actions to take to remedy a problem.

You might choose to disable some or all of the Tivoli Workload Scheduler/NetView traps for the following reasons:
1. To reduce network traffic
2. To prevent NetView user’s from receiving logged events that are not relevant

For more information about the Unison Software’s enterprise-specific traps and their variables, see “Re-configuring enterprise-specific traps” on page 190.

Tivoli Workload Scheduler/NetView configuration files
On each managed node (each node running a Tivoli Workload Scheduler/NetView agent), the selection of events and how they are reported is controlled by setting variables in two configuration files:

- The $mEvents configuration file controls the reporting of job scheduling events (101-252 in Table 37) by the mailman and batchman production processes. These events are passed on to the agent, which might convert them to SNMP traps, depending on the settings in its configuration file.
Integration with Tivoli NetView

- The MAgent configuration file controls reporting by the Tivoli Workload Scheduler/NetView agent, magent. Events selected in this file are turned into SNMP traps, which are passed to NetView by the Tivoli Workload Scheduler/NetView manager, mdemon, on the management node. The traps can also be processed by other network management systems.

**The BmEvents configuration file**
The BmEvents configuration file is named `<TWS_home>/BmEvents.conf`. Use it to configure Tivoli Workload Scheduler production processes on each workstation that has an agent installed. Its contents are described below.

```
# comment
A comment line.

OPTIONS=MASTER|OFF
If the value is set to MASTER then all job scheduling events gathered by that workstation are reported. If that workstation is the master domain manager or the backup master domain manager with full status on, then all scheduling events from the scheduling environment are reported. If the value is set to OFF, no job scheduling events are reported from that workstation. If commented, it defaults to MASTER on the master domain manager workstation, while it allows to report all job scheduling events regarding that workstation only on a workstation different from the master domain manager.

EVENT=n [n ...]
The list of events to be reported. Event numbers must be separated by at least one space. If omitted, the events reported by default are:
51 101 102 105 151 152 155 201 202 203 204 251 252

Event 51 causes mailman and batchman to report the fact that they were restarted. Events 1, 52, and 53 are not valid in this file (see "The MAgent configuration file" on page 187).

If the EVENT parameter is included, it completely overrides the defaults. To remove only event 102 from the list, for example, you must enter the following:
EVENT=51 101 105 151 152 155 201 202 203 204 251 252

See Table 37 for a description of events.

PIPE=filename
If set, job scheduling events are written to a FIFO file. To have events sent to the Tivoli Workload Scheduler/NetView agent, the setting must be:
PIPE=MAGENT.P

CHSCHED=HIGH|LOW
When set to HIGH, batchman sends an event for any schedule status transaction. When set to LOW, batchman only tracks the initial schedule status transactions. For the lifetime of schedule jobs no change of status is reported until the final state is reached. When a job has more than one final state, an event is sent for each. For example, a schedule completes with an ABEND state and event 151 is sent (schedule abended). The job is then rerun and completes successfully. The schedule is completed with a SUCC state and event 154 is sent (schedule completed). The default is HIGH. Table 38 on page 187 lists the events that are filtered by CHSCHED when it is set to LOW.

---

**IBM Tivoli Workload Scheduler Planning and Installation Guide**

186
### Table 38. Events filtered by CHSCHED

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Filtered on LOW</th>
</tr>
</thead>
<tbody>
<tr>
<td>151</td>
<td>Schedule abended</td>
<td>NO</td>
</tr>
<tr>
<td>152</td>
<td>Schedule is stuck</td>
<td>NO</td>
</tr>
<tr>
<td>153</td>
<td>Schedule started</td>
<td>YES</td>
</tr>
<tr>
<td>154</td>
<td>Schedule ended</td>
<td>NO</td>
</tr>
<tr>
<td>155</td>
<td>Until time expired onuntil = suppr</td>
<td>NO</td>
</tr>
<tr>
<td>156</td>
<td>Schedule submitted</td>
<td>YES</td>
</tr>
<tr>
<td>157</td>
<td>Schedule cancelled</td>
<td>NO</td>
</tr>
<tr>
<td>158</td>
<td>Schedule ready</td>
<td>YES</td>
</tr>
<tr>
<td>159</td>
<td>Schedule hold</td>
<td>YES</td>
</tr>
<tr>
<td>160</td>
<td>Schedule extrn</td>
<td>YES</td>
</tr>
<tr>
<td>161</td>
<td>Schedule is cancel pending</td>
<td>NO</td>
</tr>
<tr>
<td>162</td>
<td>Schedule properties changed</td>
<td>YES</td>
</tr>
<tr>
<td>163</td>
<td>Schedule is late</td>
<td>NO</td>
</tr>
<tr>
<td>164</td>
<td>Until time expired onuntil = continue</td>
<td>NO</td>
</tr>
<tr>
<td>165</td>
<td>Until time expired onuntil = cancel</td>
<td>NO</td>
</tr>
</tbody>
</table>

A `BmEvents` configuration file is included with the Tivoli Workload Scheduler software. It contains several comment lines, and a single parameter setting:

```
PIPE=MAGENT.P
```

This causes events to be reported as follows:

- If installed on the master, it will report all job scheduling events (101-252) for all workstations in the network. If installed on any other workstation, no job scheduling events will be reported. The process restart event (51) is reported regardless of the workstation type.
- The following events are reported:
  
  51 101 102 105 151 152 155 201 202 203 204 251 252

- Event information is written to a FIFO file named `MAGENT.P`, which is read by the Tivoli Workload Scheduler/NetView agent.

### The MAgent configuration file

The MAgent configuration file is named `<TWShome>/MAgent.conf`. Use it to configure the agent on each workstation. Its contents are described below.

```
# comment
    A comment line.

OPTIONS=MASTER|OFF
    If set to MASTER, the agent on this workstation sends the job scheduling events read from the `MAGENT.P` file as SNMP traps. If set to OFF, no job scheduling traps are generated by this workstation. If omitted, it defaults to MASTER on the master, and OFF on other workstations.

    This variable is required only if the master will not be used to generate job scheduling traps for the network. For example, if the master is not a
```

Appendix A. Integration with other IBM Tivoli products  187
Integration with Tivoli NetView

managed node (no agent is installed), you should set this variable to MASTER on a backup master that has an agent installed.

EVENT= n [ n ...]
The list of events to be sent as SNMP traps. With the exception of events 1, 52, and 53, traps will not be generated unless the corresponding events are turned on in the BmEvents configuration file. Event numbers must be separated by at least one space. If omitted, the events sent as traps by default are:

1 52 53 54 101 102 105 151 152 155 201 202 203 204 252

Event 1 (magent restarted) cannot be turned off.

If this parameter is included, it completely overrides the defaults. To remove only event 102 from the list, for example, you must enter the following:

EVENT=1 52 53 54 101 105 151 152 155 201 202 203 204 252

See Table 37 on page 184 for a description of events.

+name [pidfilename]
By default, the list of processes monitored by the Tivoli Workload Scheduler/NetView agent contains the following processes: magent, netman, mailman, batchman, jobman, all mailman servers, all writers, and all extended agent connections. Use this syntax to add processes to the list. If it is not a Tivoli Workload Scheduler process, you must include its PID file name. Some examples are:

+SENDMAIL /etc/sendmail.pid
+SYSLOG /etc/syslogd.pid

-name
Use this syntax to remove processes from the list of monitored processes.
To remove writer processes, use this form:

- cpuid :writer

For example, to remove the writers for all workstations with ids starting with SYS, enter:

-SYS@:WRITER

To remove all writers, enter:

-@:WRITER

To remove mailman servers 5 and A, enter:

-SERVER$5
-SERVERA

To remove all mailman servers, enter:

-SERVER@

An MAgent configuration file is included with the Tivoli Workload Scheduler/NetView software. It contains only comment lines with no parameters set. This causes SNMP traps to be generated as follows:

- If installed on the master, traps are generated for job scheduling events (101-252) on all workstations in the network. If installed on any other workstation, no job scheduling traps are generated.
- The following events result in SNMP traps:

1 52 53 54 101 102 105 151 152 155 201 202 203 204 252
• The following processes are monitored: magent, netman, mailman, batchman, jobman, all mailman servers, all writers, and all extended agent connections.

**Monitoring writers and servers**

writer and mailman server processes are started and stopped when workstations are linked and unlinked. Their transitory nature and the resulting number of status changes in NetView can cause confusion, particularly in large networks where linking and unlinking is common. For this reason, you can remove writer and mailman server processes from the list of monitored processes.

**Tivoli Workload Scheduler/NetView configuration options**

Tivoli Workload Scheduler/NetView submaps, symbols, and objects can be modified like others in NetView. The following topics describe some specific configuration options for Tivoli Workload Scheduler/NetView.

**Agent scan rate**

By default, the agents scan and update the status of their monitored processes every 60 seconds. To change the rate:

1. Login on the managed node and edit the file `<TWShome>/StartUp`.
2. Add the `-timeout` option to the magent command line.

For example, to change the rate to 120 seconds, make the following change:

```bash
<TWShome>/bin/magent -peers hosts -timeout 120
```

**Manager polling rate**

The mdemon manager polls its agents to retrieve status information about the managed nodes. The rate is defined in the file `/usr/OV/lrf/Mae.mgmt.lrf` on the management node. Unless otherwise specified, the polling rate defaults to 60 seconds.

To change the rate:

1. Edit the file to add the `-timeout` option to the mdemon command line. For example, to change the rate to 120 seconds, make the following change:

```bash
Unison_Software_Maestro_Manager: <TWShome>/bin/mdemon:
OVs_YES_START:pmd,ovwdb:-pmd,-timeout,120:OVs_WELL_BEHAVED
```
2. After making a change, delete the old registration by running the `ovdelobj` command.
3. Register the manager by running the `ovaddobj` command and supplying the name of the `lrf` file.

For more information, review the man pages for `ovaddobj(8)` and `lrf(4)`. See also *Configuring agents in NetView*

**Configuring agents in NetView**

To change the configuration of Tivoli Workload Scheduler/NetView agents in NetView, follow these steps:

1. Move down the IP Internet tree to the IP Segment submap showing all the nodes.
2. Select a node where a Tivoli Workload Scheduler/NetView agent is running. Press `Ctrl-O` to open the Object Description panel.
3. On the Object Description panel, select **Maestro - Unison Software** from the **Object Attributes** list.
4. Click the **View/Modify Object Attributes** button.
5. On the Attributes for Object panel:
Integration with Tivoli NetView

a. To ignore this agent altogether, click **False** under **Does a Maestro agent exist on this cpu?**.

b. To change the rate at which mdemon polls this agent, enter the number of seconds under **Enter the number of seconds between polling**. If this number is other than zero, it overrides the rate defined for the mdemon process (see “Manager polling rate” on page 189).

c. Click **Verify**, and then **OK** to close the Attributes for Object panel.

6. Click **OK** to close the Object Description panel.

**Configuring workstation status in NetView**

To modify the way status is indicated for a Tivoli Workload Scheduler workstation symbol, follow these steps:

1. Select a workstation symbol on the Tivoli Workload Scheduler network submap.

2. Press **Ctrl-O** to open the Object Description panel.

3. On the Object Description dialog, select **Tivoli Workload Scheduler** from the **Object Attributes** list.

4. Click **View/Modify Object Attributes**.

5. On the Attributes for Object dialog, click **True** or **False** to ignore or recognize the various job scheduling events. For example, to ignore job abend events, click **True** under **Tivoli Workload Scheduler should ignore JobAbend Events**.

6. Click **Verify**, and then **OK** to close the Attributes for Object panel.

7. Click **OK** to close the Object Description panel.

**Unison software MIB**

For a complete listing of the Unison Software enterprise MIB, see the file `TwShome/OV/Maestro.mib`.

**Re-configuring enterprise-specific traps**

The Tivoli Workload Scheduler/NetView enterprise-specific traps are configured with default messages that will serve most users’ needs. To re-configure the traps, choose **Event Configuration** from the **Options** menu. For instructions, refer to your NetView documentation or online help. It might also be helpful to review the man page for `trapd.conf(4)`.

The enterprise-specific traps and their positional variables are listed in **Table 39**.

<table>
<thead>
<tr>
<th>Trap Identifier</th>
<th>Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 *</td>
<td>uTtrapReset</td>
</tr>
<tr>
<td>51 52 * 53 *</td>
<td>uTtrapProcessReset uTtrapProcessGone uTtrapProcessAbend</td>
</tr>
</tbody>
</table>

**Table 39** lists enterprise-specific traps.
# Table 39. Enterprise-specific traps (continued)

<table>
<thead>
<tr>
<th>Trap</th>
<th>Identifier</th>
<th>Positional variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>54 *</td>
<td>uTrapXagentConnLost</td>
<td>1 Program name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 Tivoli Workload Scheduler message string, if any.</td>
</tr>
<tr>
<td>101 *</td>
<td>uTrapJobAbend</td>
<td>1 Workstation name of the schedule.</td>
</tr>
<tr>
<td></td>
<td>uTrapJobFailed</td>
<td>2 Schedule name.</td>
</tr>
<tr>
<td></td>
<td>uTrapJobLaunch</td>
<td>3 Job name. For jobs submitted with <em>at or batch</em>, if the name supplied by the user is not unique, this is the Tivoli Workload Scheduler-generated name, and the name supplied by the user appears as variable 7.</td>
</tr>
<tr>
<td></td>
<td>uTrapJobDone</td>
<td></td>
</tr>
<tr>
<td></td>
<td>uTrapJobUntil</td>
<td></td>
</tr>
<tr>
<td></td>
<td>uTrapJobRerunPrompt</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4 Workstation name on which the job runs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5 Job number (pid).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6 Job state, indicated by an integer: 1 (ready), 2 (hold), 3 (exec), 5 (abend), 6 (succ), 7 (cancl), 8 (done), 13 (fail), 16 (intro), 23 (abenp), 24 (succp), 25 (pend).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7 Job’s submitted (real) name. For jobs submitted with <em>at or batch</em>, this is the name supplied by the user if not unique. The unique name generated by Maestro appears as variable 3.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8 User name under which the job runs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9 Name of the job’s script file, or the command it executes. White space is replaced by the octal equivalent; for example, a space appears as 040.</td>
</tr>
</tbody>
</table>

Integration with Tivoli NetView

Appendix A. Integration with other IBM Tivoli products  191
### Table 39. Enterprise-specific traps (continued)

<table>
<thead>
<tr>
<th>Trap Identifier</th>
<th>Positional variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>101 * 102 * 103 104 105 * 204 *</td>
<td>uTrapJobAbend, uTrapJobFailed, uTrapJobLaunch, uTrapJobDone, uTrapJobUntil, uTrapJobRerunPrompt</td>
</tr>
<tr>
<td>10 The rate at which an <em>every</em> job runs, expressed as <em>hhmm</em>. If <em>every</em> was not specified for the job, this is -32768.</td>
<td></td>
</tr>
<tr>
<td>11 Job recovery step, indicated by an integer: 1 (stop), 2 (stop after recovery job), 3 (rerun), 4 (rerun after recovery job), 5 (continue), 6 (continue after recovery job), 10 (this is the rerun of the job), 20 (this is the run of the recovery job).</td>
<td></td>
</tr>
<tr>
<td>12 An event timestamp, expressed as: <em>yyyymmddhhmms</em> 00 (that is, year, month, day, hour, minute, second, hundredths always zeroes).</td>
<td></td>
</tr>
<tr>
<td>13 The prompt number, or zero if there is no prompt.</td>
<td></td>
</tr>
<tr>
<td>14 The prompt text, or Tivoli Workload Scheduler error message.</td>
<td></td>
</tr>
<tr>
<td>151 * 152 * 153 154 155 *</td>
<td>uTrapSchedAbend, uTrapSchedStuck, uTrapSchedStart, uTrapSchedDone, uTrapSchedUntil</td>
</tr>
<tr>
<td>1 Workstation name of the schedule.</td>
<td></td>
</tr>
<tr>
<td>2 Schedule name.</td>
<td></td>
</tr>
<tr>
<td>3 Schedule state, indicated by an integer: 1 (ready), 2 (hold), 3 (exec), 4 (stuck), 5 (abend), 6 (succl), 7 (cancl).</td>
<td></td>
</tr>
<tr>
<td>4 Tivoli Workload Scheduler error message, if any.</td>
<td></td>
</tr>
<tr>
<td>201 *</td>
<td>uTrapGlobalPrompt</td>
</tr>
<tr>
<td>1 Prompt name.</td>
<td></td>
</tr>
<tr>
<td>2 Prompt number.</td>
<td></td>
</tr>
<tr>
<td>3 Prompt text.</td>
<td></td>
</tr>
<tr>
<td>202 *</td>
<td>uTrapSchedPrompt</td>
</tr>
<tr>
<td>1 Workstation name of the schedule.</td>
<td></td>
</tr>
<tr>
<td>2 Schedule name.</td>
<td></td>
</tr>
<tr>
<td>3 Prompt number.</td>
<td></td>
</tr>
<tr>
<td>4 Prompt text.</td>
<td></td>
</tr>
</tbody>
</table>
### Table 39. Enterprise-specific traps (continued)

<table>
<thead>
<tr>
<th>Trap</th>
<th>Identifier</th>
<th>Positional variables</th>
</tr>
</thead>
</table>
| 203 *   | uTrapJobPrompt        | 1 Workstation name of the schedule.  
                                      2 Schedule name.  
                                      3 Job name.  
                                      4 Workstation name of the job.  
                                      5 Prompt number.  
                                      6 Prompt text. |
| 251 *   | uTrapLinkDropped      | 1 The to workstation name.  
                                      2 Link state, indicated by an integer: 1 (unknown), 2 (down due to an unlink), 3 (down due to an error), 4 (up).  
                                      3 Tivoli Workload Scheduler error message.  
                                      uTrapLinkDropped corresponds to link state 2) only. |
| 252 *   | uTrapLinkBroken       | 1 The to workstation name.  
                                      2 Link state, indicated by an integer: 1 (unknown), 2 (down due to an unlink), 3 (down due to an error), 4 (up).  
                                      3 Tivoli Workload Scheduler error message.  
                                      uTrapLinkBroken corresponds to link state 3) only. |

* These traps are enabled by default.

Link states 1) unknown and 4) up are not used, as they are not relative to traps 251 and 252.

---

**Tivoli Workload Scheduler/NetView program reference**

The following information is provided for those who want to run the Tivoli Workload Scheduler/NetView programs manually. The manager program, mdemon, is normally started with NetView as part of the ovstart sequence, and its run options are included in the /usr/0V/lrf/Mae.mgmt.lrf file. The agent program, magent, is normally started within the Tivoli Workload Scheduler StartUp script (<TWShome>/bin/StartUp).
Integration with Tivoli NetView

**mdemon synopsis**

`mdemon [-timeout secs] [-pmd] [-port port] [-retry secs]`

where,

- **-timeout**  
  The rate at which agents are polled, expressed in seconds. The default is 60 seconds. See "Manager polling rate" on page 189 and "Configuring agents in NetView" on page 189 for more information about changing the rate.

- **-pmd**  
  This option causes mdemon to run under NetView pmd (Port Map Demon). Otherwise, it must be run manually. This option is included by default in the file `/usr/OV/lrf/Mae.mgmt.lrf` file.

- **-port**  
  For HP-UX agents only. This identifies the port address on the managed nodes on which the HP-UX agents will respond. The default is 31112.

- **-retry**  
  The period of time mdemon waits before trying to reconnect to a non-responding agent. The default is 600 seconds.

**magent synopsis**

The syntax of magent is:

`magent -peers host [, host [...]] [-timeout secs] [-notraps] [-port port]`

where,

- **-peers**  
  For HP-UX agents only. This defines the hosts (names or IP addresses) to which the agent will send its traps. The default is 127.0.0.1 (loopback).

  For AIX agents, the `/etc/snmpd.conf` file must be modified to define the hosts to which the agent will send its traps. To add another host, for example, duplicate the existing trap line and change the host name:

  ```
  # This file contains Tivoli Workload Scheduler
  # agent registration.
  #
  trap   public host1   1.3.6.1.4.1.736 fe
  trap   public host2   1.3.6.1.4.1.736 fe
  ```

- **-timeout**  
  The rate at which the agent checks its monitored processes, expressed in seconds. The default is 60 seconds.

- **-notraps**  
  If included, the agent will not generate traps.

- **-port**  
  For HP-UX agents only. This defines the port address on which this agent responds. The default is 31112.
Integration with Tivoli Enterprise Console

This section describes integration with Tivoli Enterprise Console. It is divided into the following subsections:

- "Configuring the Tivoli Enterprise Console adapter"
- “Configuring the Tivoli Enterprise Console server” on page 196
- “Event formats” on page 197
- “Job scheduling events format” on page 213

Configuring the Tivoli Enterprise Console adapter

This subsection describes how to enable Tivoli Enterprise Console to receive Tivoli Workload Scheduler events.

The Tivoli Enterprise Console logfile adapter is used to relay events from the workstations in the scheduling environment to the Tivoli Enterprise Console event server. Depending on the workstation in your scheduling environment where you decide to install and configure the Tivoli Enterprise Console logfile adapter you can have different events displayed in the event console.

When you have installed the Tivoli Enterprise Console logfile adapter on a workstation, a set of configuration steps must be performed to enable that adapter to manage the job scheduling events. For information on how to install the Tivoli Enterprise Console logfile adapter, refer to the IBM Tivoli Enterprise Console Installation Guide.

You use the config_teclogadapter script to configure the Tivoli Enterprise Console adapter installed on the Tivoli Workload Scheduler system that you want to monitor. Perform the following steps:

1. Set the environment variables for the Tivoli endpoint by running the lcf_env script.
2. Run the config_teclogadapter script to configure the adapter. For example:

   ```bash
   config_teclogadapter [-tme] PATH [Adapter ID] [TWS Installation Path]
   ```

   where:

   - **-tme** The Tivoli Enterprise Console adapter is a TME® adapter.
   - **PATH** Specify the Tivoli Enterprise Console adapter directory when you did not specify the -tme option. Otherwise it is the endpoint directory.
   - **Adapter ID** Specify the Tivoli Enterprise Console Adapter identifier (only for Tivoli Enterprise Console 3.9 and later). If you do not specify an ID, it is ignored.
   - **TWS Installation Path** Specify the path where the Tivoli Workload Scheduler you want to monitor is installed.

The script performs the following configuration steps:

1. If no Tivoli Workload Scheduler installation path was specified, derives the home directory where it is installed.
2. Copies the config/BmEvents.conf into the home directory if it does not already exist.
Configuring the Tivoli Enterprise Console adapter

3. Configures the config/BmEvents.conf adding the list of events if not already specified and defines the event.log file as an event output.

4. Configures the configuration file of the Tivoli Enterprise Console adapter to read from the event.log file.

5. Appends the maestro.fmt file to the format file of the Tivoli Enterprise Console adapter and regenerates the cds file.

6. Restarts the Tivoli Enterprise Console adapter.

After you run the script, perform a **conman stop** and **conman start** to apply the changes.

Configuring the Tivoli Enterprise Console server

As well as configuring the Tivoli Enterprise Console adapter, you need to configure the Tivoli Enterprise Console server.

You use the **config_tecserver** script to configure the Tivoli Enterprise Console server to enable the server to receive events from the Tivoli Enterprise Console adapter. It must be run on the system where the Tivoli Enterprise Console Server is installed or on a ManagedNode of the same TME network. On the Windows platform, a TME bash is required to run the script. For example:

```
config_tecserver.sh { -newrb <RuleBase name=""> <RuleBase Path=""> -clonerb <RuleBase name=""> | -userb <RuleBase name=""> }
<EventConsole> [TECUIServer host] USER PASSWORD
```

where:

- **newrb** Specify a new RuleBase with the specified name and path.
- **clonerb** Specify the rule base to be cloned into the new RuleBase.
- **userb** Customize an already existing RuleBase.

**EventConsole** Specify the EventConsole to be created and configured.

**TECUIServer host** Specify the host name where the Tivoli Enterprise Console UI server is installed.

**USER PASSWORD** Specify the user name and password needed to access the EventConsole.

The script performs the following configuration steps:

1. If specified, creates the new RuleBase from the cloned one.
2. Adds the Tivoli Workload Scheduler baroc events definition to the specified RuleBase.
3. Adds the Tivoli Workload Scheduler rules to the RuleBase.
4. Compile the RuleBase.
5. Put the RuleBase as the Active RuleBase.
6. Configures the specified EventConsole with Tivoli Workload Scheduler filters.
7. Restarts the Tivoli Enterprise Console server.

196 IBM Tivoli Workload Scheduler Planning and Installation Guide
**Event formats**

Table 40 lists the engine event formats.

<table>
<thead>
<tr>
<th>Event</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>mstReset</td>
<td>1</td>
</tr>
<tr>
<td>mstProcessGone</td>
<td>52</td>
</tr>
<tr>
<td>mstProcessAbend</td>
<td>53</td>
</tr>
<tr>
<td>mstXagentConnLost</td>
<td>54</td>
</tr>
<tr>
<td>mstJobAbend</td>
<td>101</td>
</tr>
<tr>
<td>mstJobFailed</td>
<td>102</td>
</tr>
<tr>
<td>mstJobLaunch</td>
<td>103</td>
</tr>
<tr>
<td>mstJobDone</td>
<td>104</td>
</tr>
<tr>
<td>mstJobUntil</td>
<td>105</td>
</tr>
<tr>
<td>mstJobSubmit</td>
<td>106</td>
</tr>
<tr>
<td>mstJobCancel</td>
<td>107</td>
</tr>
<tr>
<td>mstJobReady</td>
<td>108</td>
</tr>
<tr>
<td>mstJobHold</td>
<td>109</td>
</tr>
<tr>
<td>mstJobRestart</td>
<td>110</td>
</tr>
<tr>
<td>mstJobCant</td>
<td>111</td>
</tr>
<tr>
<td>mstJobSuccp</td>
<td>112</td>
</tr>
<tr>
<td>mstJobExtn</td>
<td>113</td>
</tr>
<tr>
<td>mstJobIntro</td>
<td>114</td>
</tr>
<tr>
<td>mstJobStuck</td>
<td>115</td>
</tr>
<tr>
<td>mstJobWait</td>
<td>116</td>
</tr>
<tr>
<td>mstJobWaitd</td>
<td>117</td>
</tr>
<tr>
<td>mstJobSched</td>
<td>118</td>
</tr>
<tr>
<td>mstJobModify</td>
<td>119</td>
</tr>
<tr>
<td>mstJobLate</td>
<td>120</td>
</tr>
<tr>
<td>mstJobUntilCont</td>
<td>121</td>
</tr>
<tr>
<td>mstJobUntilCanc</td>
<td>122</td>
</tr>
<tr>
<td>mstSchedAbend</td>
<td>151</td>
</tr>
<tr>
<td>mstSchedStuck</td>
<td>152</td>
</tr>
<tr>
<td>mstSchedStart</td>
<td>153</td>
</tr>
<tr>
<td>mstSchedDone</td>
<td>154</td>
</tr>
<tr>
<td>mstSchedUntil</td>
<td>155</td>
</tr>
<tr>
<td>mstSchedSubmit</td>
<td>156</td>
</tr>
<tr>
<td>mstSchedCancel</td>
<td>157</td>
</tr>
<tr>
<td>mstSchedReady</td>
<td>158</td>
</tr>
<tr>
<td>mstSchedHold</td>
<td>159</td>
</tr>
<tr>
<td>mstSchedExtn</td>
<td>160</td>
</tr>
<tr>
<td>mstSchedCnpend</td>
<td>161</td>
</tr>
</tbody>
</table>
Configuring the Tivoli Enterprise Console adapter

Table 40. Tivoli Workload Scheduler engine event formats (continued)

<table>
<thead>
<tr>
<th>Event</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>mstSchedModify</td>
<td>162</td>
</tr>
<tr>
<td>mstSchedLate</td>
<td>163</td>
</tr>
<tr>
<td>mstSchedUntilCont</td>
<td>164</td>
</tr>
<tr>
<td>mstSchedUntilCanc</td>
<td>165</td>
</tr>
<tr>
<td>mstGlobalPrompt</td>
<td>201</td>
</tr>
<tr>
<td>mstSchedPrompt</td>
<td>202</td>
</tr>
<tr>
<td>mstJobPrompt</td>
<td>203</td>
</tr>
<tr>
<td>mstJobReovPrompt</td>
<td>204</td>
</tr>
<tr>
<td>mstLinkDropped</td>
<td>251</td>
</tr>
<tr>
<td>mstLinkBroken</td>
<td>252</td>
</tr>
<tr>
<td>mstDomainMgrSwitch</td>
<td>301</td>
</tr>
</tbody>
</table>

Positional event variables
This subsection defines the positional event variables.

Table 41. Positional variables for events 101-118,120-122,204 (job events)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule id</td>
</tr>
<tr>
<td>4</td>
<td>job name</td>
</tr>
<tr>
<td>5</td>
<td>job cpu</td>
</tr>
<tr>
<td>6</td>
<td>job number</td>
</tr>
<tr>
<td>7</td>
<td>job status</td>
</tr>
<tr>
<td>8</td>
<td>real name (different from job name only for MPE jobs)</td>
</tr>
<tr>
<td>9</td>
<td>job user</td>
</tr>
<tr>
<td>10</td>
<td>jcl name (script name or command name)</td>
</tr>
<tr>
<td>11</td>
<td>every time</td>
</tr>
<tr>
<td>12</td>
<td>recovery status</td>
</tr>
<tr>
<td>13</td>
<td>time stamp (yyyyymmddhhmm0000)</td>
</tr>
<tr>
<td>14</td>
<td>message number (not equal to zero only for job recovery prompts)</td>
</tr>
<tr>
<td>15</td>
<td>eventual text message (delimited by <code>	</code>)</td>
</tr>
<tr>
<td>16</td>
<td>record number</td>
</tr>
<tr>
<td>17</td>
<td>key flag</td>
</tr>
<tr>
<td>18</td>
<td>effective start time</td>
</tr>
<tr>
<td>19</td>
<td>estimated start time</td>
</tr>
<tr>
<td>20</td>
<td>estimated duration</td>
</tr>
<tr>
<td>21</td>
<td>deadline time (epoch)</td>
</tr>
<tr>
<td>22</td>
<td>return code</td>
</tr>
</tbody>
</table>
### Table 41. Positional variables for events 101-118,120-122,204 (job events) (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>original schedule name (schedule name for schedules not (yet) carried forward)</td>
</tr>
<tr>
<td>24</td>
<td>head job record number (different from record number for rerun/every jobs)</td>
</tr>
<tr>
<td>25</td>
<td>Schedule name</td>
</tr>
<tr>
<td>26</td>
<td>Schedule input arrival time (yyyymmddhhmm00)</td>
</tr>
</tbody>
</table>

### Table 42. Positional variables for event 119 (job property modified)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule id</td>
</tr>
<tr>
<td>4</td>
<td>job name</td>
</tr>
<tr>
<td>5</td>
<td>job cpu</td>
</tr>
<tr>
<td>6</td>
<td>job number</td>
</tr>
</tbody>
</table>
| 7        | property type:  
|          | StartTime = 2,  
|          | StopTime = 3,  
|          | Duration = 4,  
|          | TerminatingPriority = 5,  
|          | KeyStatus = 6 |
| 8        | property value |
| 9        | record number |
| 10       | key flag |
| 11       | head job record number (different from record number for rerun/every jobs) |
| 12       | real name (different from job name only for MPE jobs) |
| 13       | original schedule name (schedule name for schedules not(yet) carried forward) |
| 14       | message number (not equal to zero only for job recovery prompts) |
| 15       | Schedule name |
| 16       | Schedule input arrival time (yyyymmddhhmm00) |

### Table 43. Positional variables for events 151-161, 163-165 (schedule events)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule ID</td>
</tr>
<tr>
<td>4</td>
<td>schedule status</td>
</tr>
</tbody>
</table>
### Configuring the Tivoli Enterprise Console adapter

#### Table 43. Positional variables for events 151-161, 163-165 (schedule events) (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>record number</td>
</tr>
<tr>
<td>6</td>
<td>key flag</td>
</tr>
<tr>
<td>7</td>
<td>original schedule name (schedule name for schedules not (yet) carried forward)</td>
</tr>
<tr>
<td>8</td>
<td>time stamp</td>
</tr>
<tr>
<td>9</td>
<td>Schedule name</td>
</tr>
<tr>
<td>10</td>
<td>Schedule input arrival time (yyyyMMddhhmm00)</td>
</tr>
</tbody>
</table>

#### Table 44. Positional variables for event 162 (schedule property modified)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule id</td>
</tr>
</tbody>
</table>
| 4        | property type:  
|          | StartTime = 2  
|          | StopTime = 3 |
| 5        | property value |
| 6        | record number |
| 7        | original schedule name (schedule name for schedules not (yet) carried forward) |
| 8        | time stamp |
| 9        | Schedule name |
| 10       | Schedule input arrival time (yyyyMMddhhmm00) |

#### Table 45. Positional variables for event 202 (schedule prompt)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule id</td>
</tr>
<tr>
<td>4</td>
<td>Schedule name</td>
</tr>
<tr>
<td>5</td>
<td>Schedule input arrival time (yyyyMMddhhmm00)</td>
</tr>
</tbody>
</table>

#### Table 46. Positional variables for event 203 (job prompt)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>event number</td>
</tr>
<tr>
<td>2</td>
<td>schedule cpu</td>
</tr>
<tr>
<td>3</td>
<td>schedule id</td>
</tr>
<tr>
<td>4</td>
<td>job name</td>
</tr>
<tr>
<td>5</td>
<td>job cpu</td>
</tr>
<tr>
<td>6</td>
<td>prompt number</td>
</tr>
</tbody>
</table>
Re-loading monitoring data

The **Configure Non-TME adapter** and **Configure TME adapter** commands set up the file `BmEvents.conf` in the `TWShome` directory. This configuration file determines which information the production processes (batchman and mailman) write in the `TWSHome/log_source_file` file, by default this file is the `event.log` file, and how this information is written.

You can change the name of the log file as follows:
- Modify the `FILE` field in the `BmEvents.conf` file and restart the Tivoli Workload Scheduler processes
- Modify the `LogSource` field in the `tecad_logfile.conf` file and restarting the Tivoli Enterprise Console logfile adapter.

In the `BmEvents.conf` file the `#` sign represents a comment. Remove the `#` sign to uncomment a line.

The contents of this file are also used by other Tivoli applications that manage events, that IBM Tivoli Workload Scheduler can interact with, such as IBM Tivoli NetView and IBM Tivoli Business Systems Management.

The options you can set in the `BmEvents.conf` file are described below:

**OPTIONS=MASTER|OFF**

If the value is set to `MASTER` then all job scheduling events gathered by that workstation are reported. If that workstation is the master domain manager or the backup master domain manager, with Full Status option switched on, then all scheduling events for all workstations are reported.

If the value is set to `OFF`, the job scheduling events are reported only if they relate to the workstation where the file is configured.

If commented, it defaults to `MASTER` on the master domain manager workstation, and to `OFF` on a workstation other than the master domain manager.

**LOGGING=ALL|KEY**

Disables or enables the key flag filter mechanism.

If set to `ALL` then all events from all jobs and job streams are logged.

If set to `KEY` the event logging is enabled only for those jobs and job streams that are marked as key. The key flag is used to identify the most critical jobs or job streams. To set it in the job or job stream properties use:
- The keywords `KEYSCHED` (for job streams) and `KEYJOB` (for jobs) from the Tivoli Workload Scheduler command line interface.
- The job `Is Monitored Job` check box and job stream `In Monitored Job Stream` check box from the IBM Tivoli Workload Scheduler Job Scheduling Console.
Configuring the Tivoli Enterprise Console adapter

**SYMEVNTS=YES|NO**
If set to YES it tells the production process, batchman, to report the jobs and job streams status events immediately after having generated the new production day plan. This key is valid only if LOGGING=KEY.
If set to NO, no report is given. The default value is NO.

**CHSCHED=HIGH|LOW**
Indicates which events are to be sent during the job stream lifetime.
During the lifetime of a job stream its status can change several times depending on the status of the jobs it contains.
By using the CHSCHED option you choose how the job stream status change is reported.
If you set it to HIGH, during the job stream lifetime an event is sent any time the status of the job stream changes. Because the intermediate status of the job stream can change several times, several events can be sent, each reporting a specific status change. For example, a job stream may go into the READY state several times during its running because its status is related to the status of the jobs it contains. Each time the job stream goes into the READY state, event 153 is sent.
If you set it to LOW, during the job stream lifetime until the final status is reached, only the initial job stream state transaction is tracked. In this way the network traffic of events reporting job stream status changes is heavily reduced. When the CHSCHED value is set to LOW these are the events that are sent only the first time during the job stream life time:

<table>
<thead>
<tr>
<th>Event number</th>
<th>Event Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>153</td>
<td>TWS_Schedule_Started</td>
<td>Job stream started</td>
</tr>
<tr>
<td>156</td>
<td>TWS_Schedule_Submit</td>
<td>Job stream submitted</td>
</tr>
<tr>
<td>158</td>
<td>TWS_Schedule_Ready</td>
<td>Job stream ready</td>
</tr>
<tr>
<td>159</td>
<td>TWS_Schedule_Hold</td>
<td>Job stream hold</td>
</tr>
<tr>
<td>160</td>
<td>TWS_Schedule_Extern</td>
<td>Job stream external</td>
</tr>
<tr>
<td>162</td>
<td>TWS_Schedule</td>
<td>Job stream properties changed</td>
</tr>
</tbody>
</table>

For final status of a job stream, regardless of the value set for CHSCHED, all events reporting the final status of the job stream are reported, even if the job stream has more than one final status. For example, if a job contained in the job stream completes with an ABEND state, event 151 is sent (Job stream abended). If that job is then rerun and completes successfully, the job stream completes with a SUCC state and event 154 is sent (Job stream completed).

The default value for CHSCHED is HIGH.

**EVENT=n[ n ...]**
Identifies which events to report in the log_source_file. Event numbers must be separated by at least one space. If omitted, the events reported by default are:
51 101 102 105 111 151 152 155 201 202 203 204 251 252 301
If the EVENT parameter is included, it completely overrides the defaults. To remove only event 102 from the list, for example, you must enter the following:

EVENT=51 101 105 111 151 152 155 201 202 203 204 251 252 301

**Note:** Event 51 is always reported each time mailman and batchman are restarted, regardless of the filters specified in the EVENT parameter. If you do not wish to notify this event to the TEC event console, you must manually edit the maestro.fmt file or, for Windows environments, the maestro_nt.fmt file and comment out the following section:

```
// TWS Event Log
FORMAT TWS_Reset
  1 %s %s %s*
  event_type 1
  hostname DEFAULT
  origin DEFAULT
  agent_id $1
  software_version $2
  msg PRINTF("TWS has been reset on host %s",hostname)
  severity HARMLESS
END
```

When this section is commented out, the TEC adapter will not send event 51 to the TEC event console.

**FILE=filename**

This option is used specifically when interacting with the Tivoli Enterprise Console. Set it to the path and file name of an ASCII log file. Job scheduling events are written to this ASCII log file which is truncated whenever the batchman and mailman processes are restarted, for example at the end of each production day.

or

**FILE_NO_UTF8 =filename**

Use this option instead of the FILE option when you want job scheduling events written in the local language file specified by this parameter.
Job scheduling events

After performing the configuration steps described in the "Configuring the Tivoli Enterprise Console adapter" on page 195, use the events gathered from the Tivoli Workload Scheduler log file using the Tivoli Enterprise Console logfile adapter to perform event management and correlation using the Tivoli Enterprise Console in your scheduling environment.

This section describes the events that are generated by using the information stored in the log file specified in theIBMEvents.conf configuration file stored on the system where you installed the Tivoli Enterprise Console logfile adapter.

An important aspect to be considered when configuring the integration with the Tivoli Enterprise Console using event adapters is whether to monitor only the master domain manager or every IBM Tivoli Workload Scheduler agent.

If you integrate only the master domain manager, all the events coming from the entire scheduling environment are reported because the log file on a master domain manager logs the information from the entire scheduling network. On the Tivoli Enterprise Console event server and TEC event console all events will therefore look as if they come from the master domain manager, regardless of which IBM Tivoli Workload Scheduler agent they originate from. The workstation name, job name, and job stream name are still reported to Tivoli Enterprise Console, but as a part of the message inside the event.

If, instead, you install a Tivoli Enterprise Console logfile adapter on every IBM Tivoli Workload Scheduler agent, this results in a duplication of events coming from the master domain manager, and from each agent. Creating and using a Tivoli Enterprise Console that detects these duplicated events, based on job_name, job_cpu, schedule_name, and schedule_cpu, and keeps just the event coming from the log file on the Tivoli Workload Scheduler agent, helps you to handle this problem. The same consideration also applies if you decide to integrate the backup master domain manager, if defined, because the log file on a backup master domain manager logs the information from the entire scheduling network. For information on creating new rules for the Tivoli Enterprise Console refer to the IBM Tivoli Enterprise Console Rule Builder's Guide. For information on how to define a backup master domain manager refer to IBM Tivoli Workload Scheduler: Planning and Installation Guide.

Figure 6 on page 205 describes how an event is generated. It shows the Tivoli Enterprise Console logfile adapter installed on the master domain manager. This is to ensure that all the information about the job scheduling execution across the entire scheduling environment is available inside the log file on that workstation. You can decide, however, to install the Tivoli Enterprise Console logfile adapter on another workstation in your scheduling environment, depending on your environment and business needs.
Job scheduling events

![Diagram of job scheduling events]

**Figure 6. Event Generation Flow**

The logic that is used to generate job scheduling events is the following:

- The information logged during the job scheduling process has an event number for each type of logged activity or problem.
- Each item of information marked with an event number that appears in the EVENT field of the BMWEvents.conf file is written into the log file specified in the FILE field of the BMWEvents.conf file.
- The Tivoli Enterprise Console logfile adapter reads this information inside the log file, formats it using the structure stored in the FMT file (maestro.fmt for UNIX, maestro_nt.fmt for Windows) and forwards it to the TEC event server, using the TEC gateway defined on the managed node of the Tivoli environment.
- On the TEC event server, the structure of the formatted information is checked using the information stored in the BAROC files and, if correct, is accepted. Otherwise a parsing failure is prompted.
- Once the event is accepted by the TEC event server, a check on possible predefined correlation rules or automatic responses for that event number is made using the information stored in the RLS files.
- If defined, the correlation rules and/or automatic responses are triggered and the event is sent to the TEC event console to be displayed on the defined Event Console.

For some error conditions on event informing that the alarm condition is ended is also stored in the log file and passed to the TEC event server via the Tivoli Enterprise Console logfile adapter. This kind of event is called a clearing event. It ends on the TEC event console any related problem events.

The following table describes the events and rules provided by Tivoli Workload Scheduler.

The text of the message that is assigned by the FMT file to the event is shown in **bold**. The text message is the one that is sent by the Tivoli Enterprise Console logfile adapter to TEC event server and then to the TEC event console. The percent sign (%) in the messages indicates a variable. The name of each variable follows the message between brackets.
## Job scheduling events

### Table 48. Job scheduling events

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TWS process %s has been reset on host %s</strong> (program_name, host_name)</td>
<td>TWS_Process_Reset. Tivoli Workload Scheduler daemon process reset.</td>
</tr>
<tr>
<td>Event Class</td>
<td>TWS_Process_Gone. Tivoli Workload Scheduler process gone.</td>
</tr>
<tr>
<td>Event Class</td>
<td>TWS_Process_Abend. Tivoli Workload Scheduler process abends.</td>
</tr>
<tr>
<td>Event Class</td>
<td>TWS_Job_Abend. Send job stdlist to the TWS_user.</td>
</tr>
<tr>
<td>Event Class</td>
<td>TWS_Job_Repeated_Failure.</td>
</tr>
<tr>
<td>Automated Action (UNIX only)</td>
<td>Send job stdlist to the TWS_user. Job failed, no recovery specified.</td>
</tr>
<tr>
<td>Event Description:</td>
<td>If this job had abended more than once within a 24 hour time window, send a TWS_Job_Repeated_Failure event.</td>
</tr>
</tbody>
</table>
### Appendix A. Integration with other IBM Tivoli products

#### Job scheduling events

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Automated Action (UNIX only)</th>
<th>Event Description</th>
<th>Correlation Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Job_Abend.</td>
<td>CRITICAL.</td>
<td>Send job stdlist to the TWS_user.</td>
<td>Job failed, recovery job runs, and schedule stops</td>
<td>TWS_Job_Abend.</td>
</tr>
</tbody>
</table>

- **″Job %s.%s failed, recovery job will be run then schedule %s will be stopped″ (schedule_name, job_name, schedule_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Job failed, recovery job runs, and schedule stops.
- Correlation Activity: TWS_Job_Abend.

- **″Job %s.%s failed, this job will be rerun″ (schedule_name, job_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Job failed, the job is rerun.
- Correlation Activity: TWS_Job_Abend.

- **″Job %s.%s failed, this job will be rerun after the recovery job″ (schedule_name, job_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Job failed, recovery job is run, and the job is run again.
- Correlation Activity: TWS_Job_Abend.

- **″Job %s.%s failed, continuing with schedule %s″ (schedule_name, job_name, schedule_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to user TWS_user.
- Event Description: Job failed, the schedule proceeds.
- Correlation Activity: TWS_Job_Abend.

- **″Job %s.%s failed, running recovery job then continuing with schedule %s″ (schedule_name, job_name, schedule_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Job failed, recovery job runs, schedule proceeds.
- Correlation Activity: TWS_Job_Abend.
### Job scheduling events

**Table 48. Job scheduling events (continued)**

**“Failure while rerunning failed job %s.%s” (schedule_name, job_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Rerun of abended job abends.
- Correlation Activity: If this job had abended more than once within a 24 hour time window, send a TWS_Job_Repeated_Failure event.

**“Failure while recovering job %s.%s” (schedule_name, job_name)**

- Event Class: TWS_Job_Abend.
- Event Severity: CRITICAL.
- Automated Action (UNIX only): Send job stdlist to the TWS_user.
- Event Description: Recovery job abends.
- Correlation Activity: If this job had abended more than once within a 24 hour time window, send a TWS_Job_Repeated_Failure event.

**“Multiple failures of Job %s#%s in 24 hour period” (schedule_name, job_name)**

- Event Class: TWS_Job_Repeated_Failure.
- Event Severity: CRITICAL.
- Event Description: Same job fails more than once in 24 hours.

**“Job %s.%s did not start” (schedule_name, job_name)**

- Event Class: TWS_Job_Failed.
- Event Severity: CRITICAL.
- Event Description: Job failed to start.

**“Job %s.%s has started on CPU %s” (schedule_name, job_name, cpu_name)**

- Event Class: TWS_Job_Launched.
- Event Severity: HARMLESS.
- Event Description: Job started.
- Correlation Activity: Clearing Event - Close open job prompt events related to this job.

**“Job %s.%s has successfully completed on CPU %s” (schedule_name, job_name, cpu_name)**

- Event Class: TWS_Job_Done.
- Event Severity: HARMLESS.
- Event Description: Job completed successfully.
- Correlation Activity: Clearing Event - Close open job started events for this job and auto-acknowledge this event.

**“Job %s.%s suspended on CPU %s” (schedule_name, job_name, cpu_name)**

- Event Class: TWS_Job_Suspended.
- Event Severity: WARNING.
- Event Description: Job suspended, the until time expired (default option suppress).
Table 48. Job scheduling events (continued)

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Job_Late</td>
<td>WARNING</td>
<td>Job late, the deadline time expired before the job completed.</td>
</tr>
<tr>
<td><strong>Job %s.%s is late on CPU %s</strong> (scheduler_name, job_cpu)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Job_Until_Cont.</td>
<td>WARNING</td>
<td>Job until time expired (option continue).</td>
</tr>
<tr>
<td><strong>Job %s.%s:until (continue) expired on CPU %s</strong>, schedule_name, job_name, job_cpu</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Job_Until_Canc.</td>
<td>WARNING</td>
<td>Job until time expired (option cancel).</td>
</tr>
<tr>
<td><strong>Job %s.%s:until (cancel) expired on CPU %s</strong>, schedule_name, job_name, job_cpu</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Until_Canc.</td>
<td>WARNING</td>
<td>Schedule until time expired (option cancel).</td>
</tr>
<tr>
<td><strong>Schedule %s suspended</strong>, (schedule_name)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Susp.</td>
<td>WARNING</td>
<td>Schedule suspended, the until time expired (default option suppress).</td>
</tr>
<tr>
<td><strong>Schedule %s is late</strong>, (schedule_name)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Late.</td>
<td>WARNING</td>
<td>Schedule late, the deadline time expired before the schedule completion.</td>
</tr>
<tr>
<td><strong>Schedule %s until(continue) expired</strong>, (schedule_name)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Until_Cont.</td>
<td>WARNING</td>
<td>Schedule until time expired (option continue).</td>
</tr>
<tr>
<td><strong>Schedule %s until (cancel) expired</strong>, (schedule_name)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Until_Canc.</td>
<td>WARNING</td>
<td>Schedule until time expired (option cancel).</td>
</tr>
<tr>
<td><strong>Schedule %s has failed</strong> (schedule_name)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Abend.</td>
<td>CRITICAL</td>
<td>Schedule abends.</td>
</tr>
<tr>
<td><strong>TWS Prompt Message</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Job_Recovery_Prompt.</td>
<td>WARNING</td>
<td>Job recovery prompt issued.</td>
</tr>
<tr>
<td><strong>Correlation Activity</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TWS_Schedule_Abend.</td>
<td>CRITICAL</td>
<td>Schedule abends.</td>
</tr>
<tr>
<td>If event is not acknowledged within 15 minutes, send mail to TWS_user (UNIX only).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Job scheduling events

#### Table 48. Job scheduling events (continued)

<table>
<thead>
<tr>
<th>Event Type</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>&quot;Schedule %s is stuck&quot; (schedule_name)</strong></td>
<td>Event Class: TWS_Schedule_Stuck.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Schedule stuck.</td>
</tr>
<tr>
<td><strong>&quot;Schedule %s has started&quot; (schedule_name)</strong></td>
<td>Event Class: TWS_Schedule_Started.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Schedule started.</td>
</tr>
<tr>
<td><strong>&quot;Schedule %s has completed&quot; (schedule_name)</strong></td>
<td>Event Class: TWS_Schedule_Done.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Schedule completed successfully.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Global prompt issued.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Schedule prompt issued.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Job recovery prompt issued.</td>
</tr>
<tr>
<td><strong>&quot;Comm link from %s to %s unlinked for unknown reason&quot; (hostname, to_cpu)</strong></td>
<td>Event Class: TWS_Link_Dropped.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Tivoli Workload Scheduler link to CPU dropped for unknown reason.</td>
</tr>
<tr>
<td><strong>&quot;Comm link from %s to %s unlinked via unlink command&quot; (hostname, to_cpu)</strong></td>
<td>Event Class: TWS_Link_Dropped.</td>
</tr>
<tr>
<td></td>
<td>Event Description: Tivoli Workload Scheduler link to CPU dropped by unlink command.</td>
</tr>
</tbody>
</table>
Table 48. Job scheduling events (continued)

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Link_Dropped.</td>
<td>CRITICAL.</td>
<td>Tivoli Workload Scheduler link to CPU dropped due to error.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Link_Established.</td>
<td>HARMLESS.</td>
<td>Tivoli Workload Scheduler CPU link to CPU established.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close related TWS_Link_Dropped or TWS_Link_Failed events and auto-acknowledge this event.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Link_Failed.</td>
<td>CRITICAL.</td>
<td>Tivoli Workload Scheduler link to CPU failed due to error.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tivoli Workload Scheduler domain manager switch has occurred.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Job_Launched.</td>
<td>WARNING.</td>
<td>If after a time equal to estimated duration, the job is still in exec status, a new message is generated.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Event Class</th>
<th>Event Severity</th>
<th>Event Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TWS_Job_Ready, TWS_Job_Hold</td>
<td>WARNING.</td>
<td>If the job has a deadline and the sum of job estimated start time and estimated duration is greater than the deadline time, a new message is generated.</td>
</tr>
</tbody>
</table>
### Table 48. Job scheduling events (continued)

<table>
<thead>
<tr>
<th>Job %s.%s on CPU %s, could miss its deadline. (schedule_name, job_name, job_cpu)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Event Class: TWS_Job_Ready, TWS_Job_Hold.</td>
</tr>
<tr>
<td>• Event Severity: WARNING</td>
</tr>
<tr>
<td>• Event Description: If the job has a deadline and the sum of job estimated/effective start time and estimated duration is greater than the deadline time, a new message is generated.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Start delay of Job %s.%s on CPU %s. (schedule_name, job_name, job_cpu)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Event Class: TWS_Job_Ready.</td>
</tr>
<tr>
<td>• Event Severity: WARNING</td>
</tr>
<tr>
<td>• Event Description: If the job is still in ready status, after n minutes a new message is generated. The default value for n is 10.</td>
</tr>
</tbody>
</table>

Default criteria that control the correlation of events and the automatic responses can be changed by editing the file maestro_plus.rls (in UNIX environments) or maestront_plus.rls (in Windows environments) file. These RLS files are created during the installation of Tivoli Workload Scheduler and compiled with the BAROC file containing the event classes for the Tivoli Workload Scheduler events on the TEC event server when the Setup Event Server for TWS task is run. Before modifying either of these two files, make a backup copy of the original file and test the modified copy in your sample test environment.

For example, in the last event described in the table you can change the n value, the number of seconds the job has to be in ready state to trigger a new message, by modifying the rule job_ready_open set for the TWS_Job_Ready event class.

```ruby
rule: job_ready_open :
  
  description: 'Start a timer rule for ready',
  
  event: _event of_class 'TWS_Job_Ready'
  where [
    
    status: outside ['CLOSED'],
    schedule_name: _schedule_name,
    job_cpu: _job_cpu,
    job_name: _job_name
  ],
  
  reception action: (
    set_timer(_event,600,'ready event')
  )
)
```

For example, by changing the value from 600 to 1200 in the set_timer predicates of the reception_action action, and then by recompiling and reloading the Rule Base you change from 600 to 1200 the number of seconds the job has to be in ready state to trigger a new message.

Job scheduling events format

The integration between Tivoli Workload Scheduler and Tivoli Enterprise Console (TEC) provides the means to identify and manage a set of predefined job scheduling events. These are the events that are managed using the Tivoli Enterprise Console logfile adapter installed on the scheduling workstations. These events are listed in the following table together with the values of their positional fields. These positional fields are the ones used by the FMT files to define the event structure which, once filled up with the information stored for that specific event number in the log file, is sent by the Tivoli Enterprise Console logfile adapter to the TEC event server. For additional information, refer to “Job scheduling events” on page 204.

Table 49. Events formats table

<table>
<thead>
<tr>
<th>Event Number</th>
<th>Event Class</th>
<th>Positional Fields Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>TWS_Process_Reset</td>
<td>Positional Fields for Process Reset Events/only for batchman:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. Event number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Process name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Local workstation name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Master workstation name.</td>
</tr>
</tbody>
</table>
### Job scheduling events format

<table>
<thead>
<tr>
<th>Positional Fields for Job Events:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Event number.</td>
</tr>
<tr>
<td>2. Job stream workstation name.</td>
</tr>
<tr>
<td>3. Job stream identified.</td>
</tr>
<tr>
<td>4. Job name. For jobs submitted with at or batch, if the name supplied by the user is not unique, this is the Tivoli Workload Scheduler-generated name, and the name supplied by the user appears as variable 8 below.</td>
</tr>
<tr>
<td>5. Workstation name on which the job runs.</td>
</tr>
<tr>
<td>6. Job number.</td>
</tr>
<tr>
<td>7. Job state, indicated by an integer: 1 (ready), 2 (hold), 3 (exec), 5 (abend), 6 (suc), 7 (cancl), 8 (done), 13 (fail), 16 (intro), 23 (abemp), 24 (succp), 25 (pend).</td>
</tr>
<tr>
<td>8. Job’s submitted (real) name. For jobs submitted with at or batch, this is the name supplied by the user if not unique. The unique name generated by Tivoli Workload Scheduler appears as variable 4 above.</td>
</tr>
<tr>
<td>10. Name of the job’s script file, or the command it runs. White space is replaced by the octal equivalent; for example, a space appears as \040.</td>
</tr>
<tr>
<td>11. The rate at which an “every” job runs, expressed as hhmm. If every was not specified for the job, this is -32768.</td>
</tr>
<tr>
<td>12. Job recovery status, indicated by an integer: 1 (stop), 2 (stop after recovery job), 3 (rerun), 4 (rerun after recovery job), 5 (continue), 6 (continue after recovery job), 10 (this is the rerun of the job), 20 (this is the run of the recovery job).</td>
</tr>
<tr>
<td>13. An event timestamp. This is the local time on the workstation where the job event occurred. It is expressed as: yyyy/mm/dd hh:mm:ss (that is, year, month, day, hour, minute, second, hundredths always zeros).</td>
</tr>
<tr>
<td>14. Message number (not zero only for job recovery prompts).</td>
</tr>
<tr>
<td>15. The prompt number delimited by \’\’t, or zero if there is no prompt.</td>
</tr>
<tr>
<td>16. Job record number. Identifies in the plan the record associated to the job (not for Event number 204).</td>
</tr>
<tr>
<td>17. Job keyflag: 0 (no key flag), 1 (key flag) (not for Event number 204).</td>
</tr>
<tr>
<td>18. Effective start time of the job (not for Event number 204). It has a valid time if it occurred in the event.</td>
</tr>
<tr>
<td>19. Estimated start time of the job (not for Event number 204). It has a valid time if an Estimated Start time has been provided by the user.</td>
</tr>
<tr>
<td>20. Estimated duration of the job (not for Event number 204). Time estimated by the Tivoli Workload Scheduler engine based on statistics.</td>
</tr>
<tr>
<td>21. Deadline in Epoch (not for Event number 204). It has a valid time if a deadline time has been provided by the user.</td>
</tr>
<tr>
<td>22. The prompt text, or Tivoli Workload Scheduler error message.</td>
</tr>
<tr>
<td>23. Original schedule name (for schedules not (yet) carried forward).</td>
</tr>
<tr>
<td>24. Head job record number (different from record number for rerun/every jobs).</td>
</tr>
<tr>
<td>25. Job stream name.</td>
</tr>
<tr>
<td>26. Job stream input arrival time expressed as: yyyy/mm/dd hh:mm:ss.</td>
</tr>
</tbody>
</table>

#### Table 49. Events formats table (continued)

<table>
<thead>
<tr>
<th>Positional Fields for Job Events:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Event number.</td>
</tr>
<tr>
<td>2. Job stream workstation name.</td>
</tr>
<tr>
<td>3. Job stream identified.</td>
</tr>
<tr>
<td>4. Job name. For jobs submitted with at or batch, if the name supplied by the user is not unique, this is the Tivoli Workload Scheduler-generated name, and the name supplied by the user appears as variable 8 below.</td>
</tr>
<tr>
<td>5. Workstation name on which the job runs.</td>
</tr>
<tr>
<td>6. Job number.</td>
</tr>
<tr>
<td>7. Job state, indicated by an integer: 1 (ready), 2 (hold), 3 (exec), 5 (abend), 6 (suc), 7 (cancl), 8 (done), 13 (fail), 16 (intro), 23 (abemp), 24 (succp), 25 (pend).</td>
</tr>
<tr>
<td>8. Job’s submitted (real) name. For jobs submitted with at or batch, this is the name supplied by the user if not unique. The unique name generated by Tivoli Workload Scheduler appears as variable 4 above.</td>
</tr>
<tr>
<td>10. Name of the job’s script file, or the command it runs. White space is replaced by the octal equivalent; for example, a space appears as \040.</td>
</tr>
<tr>
<td>11. The rate at which an “every” job runs, expressed as hhmm. If every was not specified for the job, this is -32768.</td>
</tr>
<tr>
<td>12. Job recovery status, indicated by an integer: 1 (stop), 2 (stop after recovery job), 3 (rerun), 4 (rerun after recovery job), 5 (continue), 6 (continue after recovery job), 10 (this is the rerun of the job), 20 (this is the run of the recovery job).</td>
</tr>
<tr>
<td>13. An event timestamp. This is the local time on the workstation where the job event occurred. It is expressed as: yyyy/mm/dd hh:mm:ss (that is, year, month, day, hour, minute, second, hundredths always zeros).</td>
</tr>
<tr>
<td>14. Message number (not zero only for job recovery prompts).</td>
</tr>
<tr>
<td>15. The prompt number delimited by \’\’t, or zero if there is no prompt.</td>
</tr>
<tr>
<td>16. Job record number. Identifies in the plan the record associated to the job (not for Event number 204).</td>
</tr>
<tr>
<td>17. Job keyflag: 0 (no key flag), 1 (key flag) (not for Event number 204).</td>
</tr>
<tr>
<td>18. Effective start time of the job (not for Event number 204). It has a valid time if it occurred in the event.</td>
</tr>
<tr>
<td>19. Estimated start time of the job (not for Event number 204). It has a valid time if an Estimated Start time has been provided by the user.</td>
</tr>
<tr>
<td>20. Estimated duration of the job (not for Event number 204). Time estimated by the Tivoli Workload Scheduler engine based on statistics.</td>
</tr>
<tr>
<td>21. Deadline in Epoch (not for Event number 204). It has a valid time if a deadline time has been provided by the user.</td>
</tr>
<tr>
<td>22. The prompt text, or Tivoli Workload Scheduler error message.</td>
</tr>
<tr>
<td>23. Original schedule name (for schedules not (yet) carried forward).</td>
</tr>
<tr>
<td>24. Head job record number (different from record number for rerun/every jobs).</td>
</tr>
<tr>
<td>25. Job stream name.</td>
</tr>
<tr>
<td>26. Job stream input arrival time expressed as: yyyy/mm/dd hh:mm:ss.</td>
</tr>
</tbody>
</table>
### Table 49. Events formats table (continued)

<table>
<thead>
<tr>
<th>Event Code</th>
<th>Event Description</th>
<th>Positional Fields for Job Property Modified Events:</th>
<th>Positional Fields for Schedule Events:</th>
<th>Positional Fields for Schedule Property Modified Events:</th>
</tr>
</thead>
<tbody>
<tr>
<td>119</td>
<td>TWS_Job</td>
<td>1. Event number.</td>
<td>1. Event number.</td>
<td>1. Event number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Workstation name on which the job runs.</td>
<td>5. Workstation name on which the job runs.</td>
<td>5. Workstation name on which the job runs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. Property type indicated by an integer: 1 (CurrEstComplete), 2 (StartTime), 3 (StopTime), 4 (Duration), 5 (TerminatingPriority), 6 (KeyStatus).</td>
<td>7. Job stream state, indicated by an integer: 1 (ready), 2 (hold), 3 (exec), 4 (stuck), 5 (abend), 6 (succ), 7 (cancl).</td>
<td>7. Property type indicated by an integer: 1 (CurrEstComplete), 2 (StartTime), 3 (StopTime), 4 (Duration), 5 (TerminatingPriority), 6 (KeyStatus).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. Record number.</td>
<td>9. Record number.</td>
<td>9. Record number.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11. Head job record number (different from record number for rerun/every jobs).</td>
<td>11. Job’s submitted (real) name. For jobs submitted with at or batch, this is the name supplied by the user if not unique. The unique name generated by Tivoli Workload Scheduler appears as variable 4 above.</td>
<td>11. Original schedule name (for schedules not (yet) carried forward).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
<td>15. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
<td>15. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
</tr>
<tr>
<td>151</td>
<td>TWS_Schedule_Abend</td>
<td>16. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
<td>16. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
<td>16. Job stream input arrival time expressed as: yyyymmddhhmm00.</td>
</tr>
<tr>
<td>152</td>
<td>TWS_Schedule_Stuck</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>153</td>
<td>TWS_Schedule_Started</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>154</td>
<td>TWS_Schedule_Done</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>155</td>
<td>TWS_Schedule_Susp</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>156</td>
<td>TWS_Schedule_Submit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>157</td>
<td>TWS_Schedule_Cancel</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>158</td>
<td>TWS_Schedule_Ready</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>159</td>
<td>TWS_Schedule_Hold</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>160</td>
<td>TWS_Schedule_Extern</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>161</td>
<td>TWS_Schedule_CnPend</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>162</td>
<td>TWS_Schedule</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>163</td>
<td>TWS_Schedule_Late</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>164</td>
<td>TWS_Schedule_Until_Cont</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>165</td>
<td>TWS_Schedule_Until_Canc</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Job scheduling events format

<table>
<thead>
<tr>
<th>Page</th>
<th>Event Type</th>
<th>Positional Fields</th>
</tr>
</thead>
</table>
| 201  | TWS_Global_Prompt | **Positional Fields for Global Prompt Events:**  
1. Event number.  
2. Prompt name.  
3. Prompt number.  
4. Prompt text. |
| 202  | TWS_Schedule_Prompt | **Positional Fields for Schedule Prompt Events:**  
1. Event number  
2. Job stream workstation name.  
4. Job stream name.  
5. Job stream input arrival time expressed as: yyyymmddhhmm00. |
| 203  | TWS_Job_Prompt | **Positional Fields for Job Prompt Events:**  
1. Event number.  
2. Job stream workstation name.  
4. Job name.  
5. Workstation name of the job.  
6. Prompt number.  
7. Prompt text.  
8. Job stream name.  
9. Job stream input arrival time expressed as: yyyymmddhhmm00. |
| 251  | TWS_Link_Dropped | **Positional Fields for Link Dropped/Broken Events:**  
1. Event number.  
2. The "to" workstation name.  
3. Link state, indicated by an integer: 1 (unknown), 2 (down due to an unlink), 3 (down due to an error), 4 (up). |
| 252  | TWS_Link_Failed |  |
| 301  | TWS_Domain_Manager_Switch | **Positional Fields for Switch Manager Events:**  
1. Event number.  
2. New manager.  
3. The domain name.  
4. Event time stamp. |
Integration with Tivoli Monitoring

You can integrate Tivoli Workload Scheduler with Tivoli Monitoring 6.1 Fix Pack 5 using data providers, that are the interfaces of the IBM Tivoli Universal Agent. They enable the collection of data from data sources, client programs, URLs, scripts, relation tables, or SNMP agents and the transfer of the collected data to the IBM Tivoli Universal Agent, an agent that monitors any collected data.

Depending on the type and on the location of the data you want to monitor, you can use a different data provider and have the following integrations:

• If you use the script data provider you can enable an integration with Tivoli Monitoring that monitors Tivoli Workload Scheduler data by running a script at regular intervals and parsing the script output to look for errors, anomalies, and threshold conditions. For details about this integration, see “Integration with IBM Tivoli Monitoring using a script data provider.”

• If you use the file data provider you can enable an integration with Tivoli Enterprise Portal, a Tivoli Monitoring component. This data provider monitors data in a sequential text file. For details about this integration, see “Integration with Tivoli Enterprise Portal using a file data provider” on page 222.

Integration with IBM Tivoli Monitoring using a script data provider

The main prerequisite for this integration is the installation of the Universal Agent on the same workstation where the Tivoli Workload Scheduler to be monitored is installed.

Tivoli Workload Scheduler installation provides you with two scripts used as data providers to configure the Universal Agent that monitors the Tivoli Workload Scheduler. For information about the operating systems supported by this integration, refer to the IBM Tivoli Monitoring 6.1 documentation.

During the installation of the Tivoli Workload Scheduler, the following configuration scripts are copied into the ITM directory of the Tivoli Workload Scheduler workstation to be monitored:

• ITMconfig.sh for UNIX and ITMconfig.cmd for Windows
  This script configures the Universal Agent to monitor the Tivoli Workload Scheduler. The syntax is the following:
  ITMconfig -ua_home_dir ua_home_dir

  where:
  
  -ua_home_dir
    Specifies the home directory of the Universal Agent. This script runs on the workstation where the Universal Agent is installed.

• ITMCreateSituations.sh for UNIX and ITMCreateSituations.cmd for Windows
  This script configures the Tivoli Enterprise Portal Console to create and display the Tivoli Workload Scheduler situations. Copy this script onto the Tivoli Enterprise Monitoring Server and run it using the following syntax:
  ITMCreateSituations -user username -password passwd -host host -ITMHome ITMHome [-TWSHome TWSHome]

  where:
Integration with IBM Tivoli Monitoring

**username**
Specifies the user ID of the IBM Tivoli Monitoring user.

**passwd**
Specifies the password of the IBM Tivoli Monitoring user.

**host**
Specifies the host name of the Tivoli Enterprise Monitoring Server.

**ITMHome**
Specifies the home directory of the Tivoli Enterprise Monitoring Server.

**TWSHome**
Specifies the home directory of the Tivoli Workload Scheduler. Use this option only if, on the workstation where the Tivoli Enterprise Portal is installed, Tivoli Workload Scheduler is also installed.

When you log on to Tivoli Enterprise Portal for the first time, you find the following situations:

![Tivoli Workload Scheduler situations](image)

*Figure 7. Tivoli Workload Scheduler situations*

After the creation of the Tivoli Workload Scheduler situations, in the Tivoli Enterprise Portal Desktop you can see the events associated to each situation:
Note: The time-to-live <TTL> value, indicating the time that the monitored data will be kept by the Tivoli Enterprise Portal, is set to 86400 seconds (default value of 1 day). To change this value, you must manually edit the universal agent metatile: ITM_TWS_84metafile_win.mdl (Windows environments) or ITM_TWS_84metafile_ux.mdl (UNIX environments). For further information refer to the Tivoli Universal Agent User’s Guide.

Resources monitored in Tivoli Workload Scheduler
The following resources are monitored by Tivoli Universal Agent:

- “Host availability”
- “Application status” on page 220
- “Tivoli Workload Scheduler used space” on page 220
- “Tivoli Workload Scheduler available disk space” on page 220
- “Page-outs” on page 221
- “Swap space available” on page 221
- “Free space” on page 221
- “Tivoli Workload Scheduler Message file size” on page 221

Host availability: The Universal Agent monitors if the parent workstation that can respond to a ping request, is available on the network. By default this check is made every hour.

The following table lists the pre-configured actions for this monitoring resource:

Table 50. Events formats table

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The parent workstation becomes unavailable</td>
<td>Send event</td>
</tr>
</tbody>
</table>
Table 50. Events formats table (continued)

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informational</td>
<td>The parent workstation becomes available</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Application status:** The Universal Agent monitors, every minute, whether a process is up and running.

By default the following scheduling processes are monitored:
- batchman
- jobman
- mailman
- monman
- netman

These are the versions of the application status monitor:
- JOBMAN for Windows
- jobman for UNIX

The following table lists the pre-configured actions for this monitoring resource:

Table 51. Events formats table

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>If one of the monitored processes becomes unavailable.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>If one of the monitored processes becomes available.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Tivoli Workload Scheduler used space:** The Universal Agent monitors, every 15 minutes, the number of megabytes used by a specific directory. The default Tivoli Workload Scheduler directories are:
- stdlist
- schedlog

The following table lists the pre-configured actions for this monitoring source:

Table 52. Events formats table

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The used space is greater than 307200 Kilobytes.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>The used space value is between 102401 and 307200 Kilobytes.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>The used space is less than or equal to 102400 Kilobytes.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Tivoli Workload Scheduler available disk space:** The Universal Agent monitors, every 15 minutes, the amount of free disk space in the Tivoli Workload Scheduler directory.
The following table lists the pre-configured actions for this monitoring resource:

**Table 53. Events formats table**

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The free disk space is less than 2 MB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>The free disk space is less than 6 MB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>The free disk space is greater than 6 MB.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Page-outs:** The Universal Agent monitors, every 5 minutes, the number of kilobytes per second paged out on a workstation.

The following table lists the pre-configured actions for this monitoring resource:

**Table 54. Events formats table**

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>Kilobytes per second paged out are greater than 100 KB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>Kilobytes per second paged out are greater than 50 KB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>Kilobytes per second paged out are less than 50 KB.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Swap space available:** The Universal Agent monitors, every 15 minutes, the amount of available swap space.

The following table lists the pre-configured actions for this monitoring source:

**Table 55. Events formats table**

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The swap space is less than 10 MB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>The swap space is less than 25 MB.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>The swap space is greater than 25 MB.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Free space:** The Universal Agent monitors, every 15 minutes, the amount of free space of the file system that mounts the Tivoli Workload Scheduler directory.

The following table lists the pre-configured actions for this monitoring source:

**Table 56. Events formats table**

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>The file system free space is less than 5%.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>The file system free space is less than 50%.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>The file system free space is greater than 90%.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

**Tivoli Workload Scheduler Message file size:** The Universal Agent monitors, every 15 minutes, the number of megabytes used by a specific message file. By default the following Tivoli Workload Scheduler files are monitored:

- Courier.msg
- Intercom.msg
Integration with IBM Tivoli Monitoring

- Mailbox.msg
- Monbox.msg
- Moncmd.msg
- Pobox files

The following table lists the pre-configured actions for this monitoring resource:

*Table 57. Events formats table*

<table>
<thead>
<tr>
<th>Severity</th>
<th>Trigger When</th>
<th>Default Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>Megabytes used by message files are greater than 90% of the message queue.</td>
<td>Send event</td>
</tr>
<tr>
<td>Warning</td>
<td>Megabytes used by message files are greater than 50% of the message queue.</td>
<td>Send event</td>
</tr>
<tr>
<td>Informational</td>
<td>Megabytes used by message files are less than 50% of the message queue.</td>
<td>Send event</td>
</tr>
</tbody>
</table>

Integration with Tivoli Enterprise Portal using a file data provider

IBM Tivoli Monitoring monitors and manages system and network applications on a variety of platforms and keeps track of the availability and performance of all parts of your enterprise. IBM Tivoli Monitoring provides reports you can use to track trends and troubleshoot problems.

Tivoli Workload Scheduler sends events IBM Tivoli Monitoring 6.1 using the File Data Provider Feature of the Tivoli Universal Agent.

Tivoli Workload Scheduler creates and writes events for all the monitored objects (jobs and job streams) into a predefined event file that is accessed by the Tivoli Universal Agent at a set interval. The Universal Agent reads the events from the event file and interprets the data using the information contained in a metafile which is provided with Tivoli Workload Scheduler and previously loaded on the Tivoli Universal Agent by a script called **TEPConfig**. For information about this script, see “Configuring the integration with Tivoli Enterprise Portal” on page 223.

The Tivoli Universal Agent stores the event information in tables that are consolidated by the Tivoli Enterprise Monitoring Server. The collected data is then retrieved and displayed by a component called Tivoli Enterprise Portal.

The following components of IBM Tivoli Monitoring 6.1 with Fix Pack 5 must be installed on one of the allowed platforms to allow the integration with Tivoli Workload Scheduler:

- A Tivoli Enterprise Monitoring Server (referred to as the monitoring server), which acts as a collection and control point for the events received from the agents.
- A Tivoli Enterprise Portal Server (referred to as the portal server) placed between the client and the monitoring server, that enables retrieval, manipulation, and analysis of data from the agents.
- A Tivoli Enterprise Portal client, with a Java-based user interface for viewing and monitoring your enterprise. Tivoli Enterprise Portal offers two modes of operation: desktop and browser.
Integration with Tivoli Enterprise Portal

- A Tivoli Universal Agent installed on the systems that you want to monitor. In your Tivoli Workload Scheduler environment, this component must be installed on the Master domain manager system. This agent collects and distributes data to the monitoring server.

The following figure describes the architecture of the integration:

![IBM Tivoli Monitoring architecture](image)

---

This section describes the integration of Tivoli Workload Scheduler with IBM Tivoli Monitoring 6.1 with Fix Pack 5 installed. It is divided into the following subsections:

- "Configuring the integration with Tivoli Enterprise Portal"
- "Displaying events on the Tivoli Enterprise Portal desktop" on page 228
- "Events displayed by Tivoli Enterprise Portal" on page 228
- "Collecting and reloading monitoring data" on page 231

### Configuring the integration with Tivoli Enterprise Portal

This section describes the steps you must perform to enable monitoring by the Tivoli Enterprise Portal.

To enable monitoring, you run the configuration script called `TEPconfig.sh` for UNIX environments or `TEPconfig.cmd` for Windows environments on the machine where the Tivoli Workload Scheduler Master Domain Manager and the Universal Agent component of IBM Tivoli Monitoring 6.1 with fix pack 5 are installed.

1. Launch the script as follows:

   ```bash
   TEPConfig -ua_home_dir <ua_home_dir> [eventfile_path <eventfile_path>]
   [-metafile_path <metafile_path>] [-APPL_NAME <APPL_NAME>] [-TTL <TTL>]
   [-SAMPLE_FACTOR <SAMPLE_FACTOR>] [-UA]
   ``

   where:
Configuring the integration with Tivoli Enterprise Portal

-ua_home_dir

Is the home directory path of the Tivoli Universal Agent installation. This parameter is mandatory.

-eventfile_path

Is the fully qualified path of the event log file. The default value is <TWS_HOME_DIR>/event.log. If the path contains blanks, you must enclose it between double and single quotes as in the following example:

-eventfile_path='c:\Program Files\IBM\TWSEvent.log'

<metafile_path>

Is the fully qualified path to the sample metafile. The default value is <TWS_HOME_DIR>/TEP/TWS4metafile_Sample.mdl. If the path contains blanks, you must enclose it between double quotes as in the following example:

-metafile_path='c:\Program Files\IBM\TWSEventfile_Sample.mdl'

<APPL_NAME>

Is the application name that will be stored in the APPL statement of the metafile. The default value is dTWS. The value specified must be between 3 and 20 alphanumeric characters including the underscore (_), dash (-), and the asterisk (*). For details about the allowed values for this parameter refer to the Tivoli Universal Agent User’s Guide.

<TTL>

Is the amount of time in seconds (Time To Live) that the monitored data will be kept by the Tivoli Enterprise Portal. This value is set in the TTL keyword of the NAME statement in the metafile. The default value is 86400 seconds. For details about the NAME statement refer to the Tivoli Universal Agent User’s Guide.

SAMPLE_FACTOR

Is the value to set for the KUMP_DP_SAMPLE_FACTOR parameter of the Universal Agent initialization file (KUMENV on Windows platforms or um.ini on UNIX platforms). The default value is 5760. It is used to calculate the sampling frequency used by the Universal Agent to access the event file. This frequency is obtained by dividing the TTL value (default of 86400) by the KUMP_DP_SAMPLE_FACTOR. If you keep the default values for both the TTL and SAMPLE_FACTOR parameters, the event file is accessed every 15 seconds. For details about the allowed values for this parameter refer to the Tivoli Universal Agent User’s Guide.

-UA

Use this parameter if you want to modify Universal Agent related parameters only (APPL_NAME, TTL or SAMPLE_FACTOR) without changing any of the existing settings in the BmEvents.conf configuration file (if present). When you specify this parameter, if a configuration file already exists, it is not affected. If it does not exist it is not created. If this parameter is specified the first time you run the TEPConfig script, you must ensure that a copy of the BmEvents.conf file is present in the Tivoli Workload Scheduler home directory, by manually copying it to the correct path.

2. Recycle the Tivoli Workload Scheduler engine. You must always perform this step in the following cases:
Configuring the integration with Tivoli Enterprise Portal

- After you run TEPConfig for the first time.
- After any run of TEPConfig in which you add or modify ua_home_dir, eventfile_path or metafile_path parameter values.

Note: If you are using special characters in job and job stream names, you need to add the following keyword to the universal agent initialization file (KUMENV on Windows or um.1ni on UNIX platforms) to ensure correct interpretation of these characters by Tivoli Enterprise Portal:

```plaintext
KUMP_DEFAULT_CODEPAGE=UTF-8
```

Restart the universal agent after this change.

The script performs the following steps:

1. Checks if a BmEvents.conf file already exists in the Tivoli Workload Scheduler home directory. If the file exists, it is renamed to BmEvents.conf.prev to maintain any existing customizations. A new copy of BmEvents.conf is made in the home directory which includes the existing customizations and the new values needed for the integration with Tivoli Enterprise Portal. If the BmEvents.conf file is not found in the Tivoli Workload Scheduler home directory, then <TWS_HOME_DIR>\config\BmEvents.conf (on Windows environments) or the<TWS_HOME_DIR>/OV/BmEvents.conf (on UNIX environments) is copied to the Tivoli Workload Scheduler home directory.

2. Adds a list of events to the configuration file in the EVENT parameter.

3. Customizes the sample metafile by setting the values for the APPL_NAME, eventfile_path, and TTL parameters, if different from the default values.

4. Performs the import or refresh of the customized metafile TWS84metafile.mdl. The Universal Agent uses this metafile to interpret the information stored in the event log and passes this monitoring data to the Tivoli Enterprise Monitoring Server.

5. If the import or refresh of the metafile was successful, an automatic restart of the Universal Agent is performed. If any warning or error messages were issued by the Universal Agent during the process, a restart is not performed. In this case, you must manually restart the Universal Agent. A restart is required in the following cases:
   - It is the first time you run the TEPConfig script, or
   - Each time you change the value of the SAMPLE_FACTOR parameter.

For a detailed description of the fields modified by the customization script, refer to "The BmEvents.conf file".

When the Tivoli Enterprise Portal integration is enabled, for each production plan the Tivoli Workload Scheduler master domain manager logs events related to jobs and job streams in the specified event file. The Tivoli Universal Agent retrieves the logged events from the event file and parses them using the installed metafile. The monitored data is then passed by the Tivoli Universal Agent to the Tivoli Enterprise Monitoring Server and to the Tivoli Enterprise Portal Server. The Tivoli Enterprise Portal Client displays the collected data.

The BmEvents.conf file: The TEPConfig script sets up the file BmEvents.conf in the TwShome directory. This configuration file determines what information the production processes (batchman and mailman) write in the TwShome/log_source_file file, by default this file is the event.log file, and how this information is written.
Configuring the integration with Tivoli Enterprise Portal

You can change the name of the log file as follows:

- Run the **TEPConfig** script specifying the `<eventfile_path>` parameter. The value you specify changes both the BmEvents.conf and the metafile information so that both the Universal Agent and Tivoli Workload Scheduler are synchronized. This is the preferred method.
- Modify the **FILE** field in the BmEvents.conf file and restart the Tivoli Workload Scheduler processes. In this case, you also have to manually change the metafile information and perform a refresh of the Universal Agent configuration to synchronize the Universal Agent and IBM Tivoli Workload Scheduler.

In the BmEvents.conf file the # sign represents a comment. Remove the # sign to uncomment a line.

The options you can set in the BmEvents.conf file are described below:

**OPTIONS=MASTER|OFF**
Determines for which workstations the gathered events are reported.
Possible values are:

**MASTER**
All job scheduling events gathered by that workstation are reported. If that workstation is the master domain manager or the backup master domain manager, with Full Status option switched on, then all scheduling events for all workstations are reported.

**OFF**
The job scheduling events are reported only if they relate to the workstation where the file is configured.

If commented, it defaults to **MASTER** on the master domain manager workstation, and to **OFF** on a workstation other than the master domain manager.

**LOGGING=ALL|KEY**
Disables or enables the key flag filter mechanism. Possible values are:

**ALL**
If set to this value, all events from all jobs and job streams are logged.

**KEY**
If set to this value, the event logging is enabled only for those jobs and job streams that are marked as key. The key flag is used to identify the most critical jobs or job streams. To set it in the job or job stream properties use:

- The keywords **KEYSCHED** (for job streams) and **KEYJOB** (for jobs) from the Tivoli Workload Scheduler command line interface.
- The job **Is Monitored Job** check box and job stream **In Monitored Job Stream** check box from the IBM Tivoli Workload Scheduler Job Scheduling Console.

The **TEPConfig** script sets the value of this parameter to **KEY**.

**SYMEVNTS=YES|NO**
Used in conjunction with **LOGGING=KEY**. It determines whether events concerning jobs and job streams are to be reported immediately after a plan creation.

**YES**
If set to this value, it tells the production process, **batchman**, to report the jobs and job streams status events immediately after having generated the new production day plan. This key is valid only if **LOGGING=KEY**. For the integration with Tivoli Enterprise
Configuring the integration with Tivoli Enterprise Portal

Portal, this value will cause a bulk discovery to be automatically performed after each new plan creation.

NO  If set to this value, no report is given.

The default value is NO.

**CHSCHED=HIGH|LOW**

Indicates which events are to be sent during the job stream lifetime.

During the lifetime of a job stream its status can change several times depending on the status of the jobs it contains. By using the CHSCHED option you choose how the job stream status change is reported.

**HIGH**  If you set this value, during the job stream lifetime an event is sent any time the status of the job stream changes. Because the intermediate status of the job stream can change several times, several events can be sent, each reporting a specific status change. For example, a job stream may go into the READY state several times during its running because its status is related to the status of the jobs it contains. Each time the job stream goes into the READY state, event 153 is sent.

**LOW**  If you set this value, during the job stream lifetime until the final status is reached, only the initial job stream state transaction is tracked. In this way the network traffic of events reporting job stream status changes is heavily reduced.

When the CHSCHED value is set to LOW these are the events that are sent only the first time during the job stream life time:

<table>
<thead>
<tr>
<th>Event number</th>
<th>Event Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>153</td>
<td>TWS_Schedule_Started</td>
<td>Job stream started</td>
</tr>
<tr>
<td>156</td>
<td>TWS_Schedule_Submit</td>
<td>Job stream submitted</td>
</tr>
<tr>
<td>158</td>
<td>TWS_Schedule_Ready</td>
<td>Job stream ready</td>
</tr>
<tr>
<td>159</td>
<td>TWS_Schedule_Hold</td>
<td>Job stream hold</td>
</tr>
<tr>
<td>160</td>
<td>TWS_Schedule_Extern</td>
<td>Job stream external</td>
</tr>
<tr>
<td>162</td>
<td>TWS_Schedule</td>
<td>Job stream properties changed</td>
</tr>
</tbody>
</table>

For final status of a job stream, regardless of the value set for CHSCHED, all events reporting the final status of the job stream are reported, even if the job stream has more than one final status. For example, if a job contained in the job stream completes with an ABEND state, event 151 is sent (Job stream abended). If that job is then rerun and completes successfully, the job stream completes with a SUCC state and event 154 is sent (Job stream completed).

The default value for CHSCHED is HIGH.

**EVENT=n[ n ...]**

Identifies which events to report in the log_source_file. Event numbers must be separated by at least one space. The events added by the configuration script are the following:

101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 120
121 122 151 152 153 154 155 156 157 158 159 160 161 163 164 165 204
Configuring the integration with Tivoli Enterprise Portal

If the EVENT parameter is manually commented out, the events read by the Tivoli Universal Agent in the event file are:

101 102 105 151 152 155 204

FILE=filename
This is the path and file name of an ASCII log file where job scheduling events are written. This parameter is set by the configuration script. This file is truncated whenever the batchman and mailman processes are restarted, for example at the end of each production day.

Displaying events on the Tivoli Enterprise Portal desktop
To display Tivoli Workload Scheduler events from your Tivoli Enterprise Portal desktop, open the default workspace provided for the TWSd application. This default view contains two tables that show Tivoli Workload Scheduler jobs and job streams.

Optionally, you can install the customized workspaces and situations by using the programs included with Tivoli Workload Scheduler.

These programs are found in the CD_ROOT directory of your installation CD. To use the customized solution, do the following:

1. On the machines where your portal server and monitoring server are installed, use one of the following installation programs, depending on the operating system:
   - setupaix.bin
   - setupwin32.exe
   - setuplinux390.bin
   - setupLinux.bin
   - setupSolaris.bin

2. Launch the installation program and follow the instructions shown on the screen. Some options are already selected and require you to click Next to proceed. In the directory window, enter the path where you have installed IBM Tivoli Monitoring, the second box is pre-filled with the location from where the installation was launched.

3. On the application selection screen select the Tivoli Workload Scheduler Customized Solution and click Next. The final verification screen opens.

4. Click the Next button to start the installation.

5. Log on to the TEP client to view the pre-defined workspaces, situations and queries.

Events displayed by Tivoli Enterprise Portal

Table 59 lists the events logged for the integration with Tivoli Enterprise Portal.

<table>
<thead>
<tr>
<th>Event</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mstJobAbend</td>
<td>101</td>
<td>Job abended</td>
</tr>
<tr>
<td>mstJobFailed</td>
<td>102</td>
<td>Job failed</td>
</tr>
<tr>
<td>mstJobLaunch</td>
<td>103</td>
<td>Job launched</td>
</tr>
<tr>
<td>mstJobDone</td>
<td>104</td>
<td>Job done</td>
</tr>
<tr>
<td>mstJobUntil</td>
<td>105</td>
<td>Job suspended until expired</td>
</tr>
<tr>
<td>mstJobSubmit</td>
<td>106</td>
<td>Job submitted</td>
</tr>
</tbody>
</table>

Table 59. Tivoli Workload Scheduler engine event formats for Tivoli Enterprise Portal
### Table 59. Tivoli Workload Scheduler engine event formats for Tivoli Enterprise Portal (continued)

<table>
<thead>
<tr>
<th>Event</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mstJobCancel</td>
<td>107</td>
<td>Job cancelled</td>
</tr>
<tr>
<td>mstJobReady</td>
<td>108</td>
<td>Job in READY status</td>
</tr>
<tr>
<td>mstJobHold</td>
<td>109</td>
<td>Job in HOLD status</td>
</tr>
<tr>
<td>mstJobRestart</td>
<td>110</td>
<td>Job restarted</td>
</tr>
<tr>
<td>mstJobCant</td>
<td>111</td>
<td>Job Failed</td>
</tr>
<tr>
<td>mstJobSuccp</td>
<td>112</td>
<td>Job Successful pending</td>
</tr>
<tr>
<td>mstJobExtn</td>
<td>113</td>
<td>Job extern</td>
</tr>
<tr>
<td>mstJobIntro</td>
<td>114</td>
<td>Job in INTRO status</td>
</tr>
<tr>
<td>mstJobStuck</td>
<td>115</td>
<td>Job in STUCK status</td>
</tr>
<tr>
<td>mstJobWait</td>
<td>116</td>
<td>Job in WAIT status</td>
</tr>
<tr>
<td>mstJobWaitd</td>
<td>117</td>
<td>Job in wait deferred status</td>
</tr>
<tr>
<td>mstJobSched</td>
<td>118</td>
<td>Job in scheduled status</td>
</tr>
<tr>
<td>mstJobLate</td>
<td>120</td>
<td>Job is late</td>
</tr>
<tr>
<td>mstJobUntilCont</td>
<td>121</td>
<td>Job UNTIL time expired with Continue option</td>
</tr>
<tr>
<td>mstJobUntilCanc</td>
<td>122</td>
<td>Job UNTIL time expired with Cancel option</td>
</tr>
<tr>
<td>mstSchedAbend</td>
<td>151</td>
<td>Job Stream abended</td>
</tr>
<tr>
<td>mstSchedStuck</td>
<td>152</td>
<td>Job stream is stuck</td>
</tr>
<tr>
<td>mstSchedStart</td>
<td>153</td>
<td>Job stream started</td>
</tr>
<tr>
<td>mstSchedDone</td>
<td>154</td>
<td>Job stream done</td>
</tr>
<tr>
<td>mstSchedUntil</td>
<td>155</td>
<td>Job Stream suspended, until time expired</td>
</tr>
<tr>
<td>mstSchedSubmit</td>
<td>156</td>
<td>Job stream submitted</td>
</tr>
<tr>
<td>mstSchedCancel</td>
<td>157</td>
<td>Job Stream cancelled</td>
</tr>
<tr>
<td>mstSchedReady</td>
<td>158</td>
<td>Job stream in READY status</td>
</tr>
<tr>
<td>mstSchedHold</td>
<td>159</td>
<td>Job stream in HOLD status</td>
</tr>
<tr>
<td>mstSchedExtn</td>
<td>160</td>
<td>Job stream extern</td>
</tr>
<tr>
<td>mstSchedCnpend</td>
<td>161</td>
<td>Job Stream in CANCEL Pending status</td>
</tr>
<tr>
<td>mstSchedLate</td>
<td>163</td>
<td>Job Stream is late</td>
</tr>
<tr>
<td>mstSchedUntilCont</td>
<td>164</td>
<td>Job Stream Until time expired with continue option</td>
</tr>
</tbody>
</table>
Displaying events on the Tivoli Enterprise Portal Desktop

Table 59. Tivoli Workload Scheduler engine event formats for Tivoli Enterprise Portal (continued)

<table>
<thead>
<tr>
<th>Event</th>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mstSchedUntilCanc</td>
<td>165</td>
<td>Job Stream until time expired with cancel option</td>
</tr>
<tr>
<td>mstJobRecovPrompt</td>
<td>204</td>
<td>Job Recovery prompt issued</td>
</tr>
</tbody>
</table>

Positional event variables: This section defines the positional event variables.

Table 60. Positional variables for events 101-118, 120-122, 204 (job events)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Event number</td>
</tr>
<tr>
<td>2</td>
<td>Job stream cpu</td>
</tr>
<tr>
<td>3</td>
<td>Job Stream id</td>
</tr>
<tr>
<td>4</td>
<td>Job name</td>
</tr>
<tr>
<td>5</td>
<td>Job CPU</td>
</tr>
<tr>
<td>6</td>
<td>Job number</td>
</tr>
<tr>
<td>7</td>
<td>Job status</td>
</tr>
<tr>
<td>8</td>
<td>Real name (different from job name only for MPE jobs)</td>
</tr>
<tr>
<td>9</td>
<td>Job user</td>
</tr>
<tr>
<td>10</td>
<td>Jcl name (script name or command name)</td>
</tr>
<tr>
<td>11</td>
<td>Every time</td>
</tr>
<tr>
<td>12</td>
<td>Recovery status</td>
</tr>
<tr>
<td>13</td>
<td>Time stamp (yyyymmddhhmm0000)</td>
</tr>
<tr>
<td>14</td>
<td>Message number (not equal to zero only for job recovery prompts)</td>
</tr>
<tr>
<td>15</td>
<td>Eventual text message (delimited by ‘\t’)</td>
</tr>
<tr>
<td>16</td>
<td>Record number</td>
</tr>
<tr>
<td>17</td>
<td>Key flag</td>
</tr>
<tr>
<td>18</td>
<td>Effective start time</td>
</tr>
<tr>
<td>19</td>
<td>Estimated start time</td>
</tr>
<tr>
<td>20</td>
<td>Estimated duration</td>
</tr>
<tr>
<td>21</td>
<td>Deadline time (epoch)</td>
</tr>
<tr>
<td>22</td>
<td>Return code</td>
</tr>
<tr>
<td>23</td>
<td>Original schedule name (schedule name for schedules not (yet) carried forward)</td>
</tr>
<tr>
<td>24</td>
<td>Head job record number (different from record number for rerun/every jobs)</td>
</tr>
<tr>
<td>25</td>
<td>Job stream name</td>
</tr>
<tr>
<td>26</td>
<td>Job stream scheduled time (yyyymmddhhmm00)</td>
</tr>
<tr>
<td>27</td>
<td>Event Version</td>
</tr>
<tr>
<td>28</td>
<td>Resource Name</td>
</tr>
<tr>
<td>29</td>
<td>Resource Type</td>
</tr>
</tbody>
</table>
Table 60. Positional variables for events 101-118, 120-122, 204 (job events) (continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Job external status</td>
</tr>
<tr>
<td>31</td>
<td>Job ID</td>
</tr>
<tr>
<td>32</td>
<td>Name space</td>
</tr>
<tr>
<td>33</td>
<td>Submitter</td>
</tr>
<tr>
<td>34</td>
<td>Job end time</td>
</tr>
<tr>
<td>35</td>
<td>Job submit time</td>
</tr>
<tr>
<td>36</td>
<td>Cpu usage</td>
</tr>
<tr>
<td>37</td>
<td>Memory usage</td>
</tr>
<tr>
<td>38</td>
<td>Operation number</td>
</tr>
<tr>
<td>39</td>
<td>Job error code</td>
</tr>
<tr>
<td>40</td>
<td>Critical job identifier</td>
</tr>
</tbody>
</table>

Table 61. Positional variables for events 151-161, 163-165 (job stream events)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Event number</td>
</tr>
<tr>
<td>2</td>
<td>Job stream CPU</td>
</tr>
<tr>
<td>3</td>
<td>Job stream ID</td>
</tr>
<tr>
<td>4</td>
<td>Job stream status</td>
</tr>
<tr>
<td>5</td>
<td>Record number</td>
</tr>
<tr>
<td>6</td>
<td>Key flag</td>
</tr>
<tr>
<td>7</td>
<td>Original job stream name (job stream name for job streams not (yet) carried forward)</td>
</tr>
<tr>
<td>8</td>
<td>Time stamp (yyyyymmddhhmm0000)</td>
</tr>
<tr>
<td>9</td>
<td>Job stream name</td>
</tr>
<tr>
<td>10</td>
<td>Job stream scheduled time (yyyyymmddhhmm00)</td>
</tr>
<tr>
<td>11</td>
<td>Event version</td>
</tr>
</tbody>
</table>

Collecting and reloading monitoring data

When the integration is enabled, the Tivoli Workload Scheduler engine starts to log monitoring events after each new plan production. To reload all monitoring data, or to obtain the initial status of your monitored objects, use the command `bulk_discovery` command. This command checks the current status of all the monitored objects (jobs and job streams) within the Symphony file, writes the corresponding events in the event.log file for retrieval by the Tivoli Enterprise Portal.

Notes:

1. If a JnextPlan is requested while a bulk_discovery is in progress, some events related to jobs and job streams might no longer be visible from the Tivoli Enterprise Portal console views when the new plan is built. Inconsistencies might occur because the JnextPlan process deletes and reallocates the event file at the end of its activity. If the Universal Agent has not read all records of the event file before the file is reallocated, it will not be aware of any missing job or jobstream status until a new change of status occurs. To ensure consistency, issue a bulk_discovery command at the end of the JnextPlan, or set...
Collecting and reloading monitoring data

SYMEVNTS=\texttt{YES} in the \texttt{BmEvents.conf} file so that a bulk discovery is performed automatically after every JnextPlan activity.

2. If you recycle the Tivoli Universal Agent, events are lost. If the Universal Agent is stopped for any reason, also as a result of a TCP/IP connection failure, events are lost unless you have saved them using the Universal Agent history function. In these cases, you issue a bulk discovery to refresh the status of all monitored objects.

For details about the syntax and output of the bulk\_discovery command, refer to the Reference Guide.

Integration with Tivoli Storage Manager

This section describes the integration of Tivoli Workload Scheduler with Tivoli Storage Manager version 5.4, an enterprise-wide storage management application that provides automated storage management services to workstations, personal computers, and file servers from a variety of vendors, with a variety of operating systems.

Integration with Tivoli Storage Manager is supported only on UNIX platforms.

Tivoli Storage Manager administrators perform several types of operations regularly each day using a built-in scheduling facility, which provides a simple mechanism to automate routine tasks. This scheduling facility, however, does not provide the ability to assign dependencies among scheduled tasks or to assign limits or priorities.

Using the extended agent, you can integrate Tivoli Storage Manager with Tivoli Workload Scheduler and assign dependencies among Tivoli Storage Manager scheduled tasks or assign limits or priorities.

You can schedule the following Tivoli Storage Manager tasks using Tivoli Workload Scheduler:

- Server administrative tasks:
  - Database backup (\texttt{BACKUP DB})
  - Volume history backup (\texttt{BACKUP VOLHISTORY})
  - Device configuration backup (\texttt{BACKUP DEVMFILE})
  - Delete volume history (\texttt{DELETE VOLHISTORY})
  - Inventory expiration (\texttt{EXPIRE INVENTORY})
- Client backup task:
  - Client backup (\texttt{DEFINE SCHEDULE} and \texttt{DEFINE ASSOCIATION})

When you install the Tivoli Workload Scheduler, the \texttt{tsmxagent.sh} script, representing the Tivoli Storage Manager access method, and its method options file, \texttt{tsmxagent.opts}, are copied to the \texttt{TWS\_HOME/methods} directory.

To schedule a job in Tivoli Workload Scheduler to run Tivoli Storage Manager commands for administrative and client backup purposes, perform the following steps:

1. Type the user ID of the administrator that accesses the Tivoli Storage Manager server in the \texttt{tsmAdmin} variable of the \texttt{tsmxagent.opts} file. The default value is \texttt{admin}. 
2. Define the password of the administrator in the TSMPASS parameter defined using the Tivoli Workload Scheduler parms utility.

3. Define an extended agent workstation that uses the tsxagent access method. This logical workstation definition must be hosted by a Tivoli Workload Scheduler physical workstation, either a Master, a Domain Manager, or a fault-tolerant agent workstation. An example is the following:

   CPUNAME TSMXAGENT
   OS UNIX
   NODE dontcare
   TCPADDR 31111
   FOR MAESTRO
   HOST TWSFTA
   ACCESS "tsxagent"
   TYPE X-AGENT
   END

   In this case, TSMXAGENT is the name of the extended agent workstation, dontcare is the node name, 31111 is the TCP address, TWSFTA is the host, and tsxagent is the name of the access method. The NODE and TCPADDR definitions are ignored.

4. Create a job for each of the following tasks:
   - To back up a Tivoli Storage Manager database to sequential access volumes, type the following command:
     `ADMIN DBBACKUP devclass=DEVCLASS [optionalParms]`
     where:
     - `DEVCLASS` Specifies the name of the sequential access device class to use for the backup.
     - `optionalParms` Specifies the optional parameters supported by the Tivoli Storage Manager.
   - To back up sequential volume history information to one or more files, type the following command:
     `ADMIN VHBACKUP [optionalParms]`
     where:
     - `optionalParms` Specifies the optional parameters supported by the Tivoli Storage Manager.
   - To back up IBM Tivoli Storage Manager device information to one or more files, type the following command:
     `ADMIN DCBACKUP [optionalParms]`
     where:
     - `optionalParms` Specifies the optional parameters supported by the Tivoli Storage Manager.
   - To delete volume history file records that are no longer needed (for example, records for obsolete database backup volumes), type the following command:
     `ADMIN DELVOLHIST type=TYPE todate=TODATE [optionalParms]`
     where:
     - `TYPE` Specifies the type of records, which also meet the date and time criteria, to delete from the volume history file. You must specify the
parameters in the order shown. For details about the possible values see the IBM Tivoli Storage Manager Administrator’s Reference guide.

TODATE
Specifies the date to use to select sequential volume history information to be deleted. Tivoli Storage Manager deletes only those records with a date on or before the date you specify. You can specify the date using the following format: MM/DD/YYYY.

optional_parms
Specifies the optional parameters supported by the Tivoli Storage Manager.

• To start inventory expiration processing, type the following command:
  ADMIN EXPIREINV [optional_parms]
  where:
  optional_parms
  Specifies the optional parameters supported by the Tivoli Storage Manager. For details, see the IBM Tivoli Storage Manager Administrator’s Reference guide.

• To schedule a client backup, type the following command:
  CLIENT DOMAIN SCHEDULE CLIENT [optional_parms]
  where:
  DOMAIN
  Specifies the name of the policy domain to which this schedule belongs.
  SCHEDULE
  Specifies the name of the schedule that you want to associate with one or more clients.
  CLIENT
  Specifies the name of a server-prompted Tivoli Storage Manager client ready for scheduling a backup.
  optional_parms
  Specifies the optional parameters supported by the Tivoli Storage Manager.

For details about optional parameters (optional_parms), see the IBM Tivoli Storage Manager Administrator’s Reference guide.

5. After submitting the jobs, on the Tivoli Storage Manager server verify that the commands ran successfully.
Integration with Tivoli License Compliance Manager

This section describes the integration of Tivoli Workload Scheduler with Tivoli License Compliance Manager version 2.3.

Integration with Tivoli License Compliance Manager is supported on all operating systems.

When you install Tivoli Workload Scheduler on a master domain manager, a backup master domain manager, a domain manager, a fault-tolerant agent, a standard agent, or an extended agent, depending on the operating system, the following signature files are copied to the TSW_HOME directory:

- ITWSS0804.SYS2 on Sun OS
- ITWSX0804.SYS2 on AIX
- ITWSH0804.SYS2 on HP-UX
- ITWSL0804.SYS2 on Linux
- ITWSW0804.SYS2 on Windows

Before updating the Tivoli License Compliance Manager catalog with these new signatures used to detect if Tivoli Workload Scheduler is installed or in use on a workstation, verify that the size of the each of these files is at least of 1 KB.
Integration with Tivoli License Compliance Manager
Appendix B. Support information

If you have a problem with your IBM software, you want to resolve it quickly. This section describes the following options for obtaining support for IBM software products:

- Using IBM Support Assistant
- Searching knowledge bases on page 238
- Obtaining fixes on page 239
- Receiving weekly support updates on page 239
- Contacting IBM Software Support on page 241

Using IBM Support Assistant

The IBM Support Assistant is a free, stand-alone application that you can install on any workstation. You can then enhance the application by installing product-specific plug-in modules for the IBM products you use.

The IBM Support Assistant saves you time searching product, support, and educational resources. The IBM Support Assistant helps you gather support information when you need to open a problem management record (PMR), which you can then use to track the problem.

The product-specific plug-in modules provide you with the following resources:
- Support links
- Education links
- Ability to submit problem management reports

The IBM Support Assistant Web site is at http://www.ibm.com/software/support/isa/. Use this site for the following:
- Obtain general information about the IBM Support Assistant
- Download and install the IBM Support Assistant application. Full instructions are provided.
- Determine if a plug-in is available for a specific product (or go direct to the plug-ins page at http://www.ibm.com/software/support/isa/plugins.html)

To locate and download the plug-in for a product, use the IBM Support Assistant's interface. Full instructions on how to use the application and plug-in are provided within the interface. For example, on version 3.0.1 of the IBM Support Assistant, click Updater, click New products and tools, expand Tivoli, select the plug-in, and click Install.

If you cannot find the solution to your problem in the IBM Support Assistant, see "Searching knowledge bases" on page 238.

Tivoli Workload Scheduler IBM Support Assistant plug-in version and upgrade issues

The IBM Tivoli Workload Scheduler plug-in for the IBM Support Assistant has not changed since version 8.3. If you have already installed it in the IBM Support Assistant you need take no further action. If you are planning to install it for the first time with version 8.4 you should be aware that the plug-in name and many
other references in the plug-in have "8.3" as the product version number. This does not mean that it will not work with version 8.4. The plug-in is fully compatible with version 8.4 and performs in exactly the same way as it does in version 8.3.

Searching knowledge bases

You can search the available knowledge bases to determine whether your problem was already encountered and is already documented.

Searching the local information center

IBM provides extensive documentation that you can install on your local computer or on an intranet server. You can use the search function of this information center to query conceptual information, instructions for completing tasks, and reference information.

The information center is included on the separate Quick Start CD available as part of the product bundle. Insert the CD in a CD drive on a Windows computer, and the information center automatically opens.

Searching the Internet

If you cannot find an answer to your question in the information center, search the Internet for the latest, most complete information that might help you resolve your problem.

To search multiple Internet resources for your product, use the Web search topic in your information center. In the navigation frame, click Troubleshooting and support ▶ Searching knowledge bases and select Web search. From this topic, you can search a variety of resources, including the following:

- IBM technote
- IBM downloads
- IBM Redbooks
- IBM developerWorks®
- Forums and newsgroups
- Google

Search the IBM support Web site

The IBM software support Web site has many publications available online, one or more of which might provide the information you require:

2. Select Tivoli under the Select a brand and/or product heading.
3. Select IBM Tivoli Workload Scheduler under Select a product
4. Click Go.
5. Under the Primary support resources heading and Learn subheading, choose from the list of different types of product support publications:
   - Information center
   - Support Technical Exchange
   - IBM Tivoli software training
   - Manuals
   - Redbooks
A search for the selected documentation type is performed, and the results displayed.

6. Use the on-screen navigation to look through the displayed list for the document you require, or use the options in the Search within results for section to narrow the search criteria. You can add Additional search terms or select a specific Document type. You can also change the sort order of the results (Sort results by). Then click to start the search.

To access some of the publications you need to register (indicated by a key icon beside the publication title). To register, select the publication you want to look at, and when asked to sign in follow the links to register yourself. There is also a FAQ available on the advantages of registering.

### Obtaining fixes

A product fix might be available to resolve your problem. To determine what fixes are available for your IBM software product, follow these steps:

2. Select Tivoli under the Select a brand and/or product heading.
3. Select IBM Tivoli Workload Scheduler under Select a product.
4. Click Go.
5. Under the Primary support resources heading and Download, subheading, either choose one of the displayed most-popular downloads, or click View all downloads. A search for the downloads is performed, and the results displayed.
6. Use the on-screen navigation to look through the displayed list for the download you require, or use the options in the Search within results for section to narrow the search criteria. You can add Additional search terms, or select a specific Download type, Platform/Operating system, and Versions, and then click to start the search.
7. Click the name of a fix to read the description of the fix and to optionally download the fix.

For more information about the types of fixes that are available, see the IBM Software Support Handbook at [http://techsupport.services.ibm.com/guides/handbook.html](http://techsupport.services.ibm.com/guides/handbook.html)

### Receiving weekly support updates

To receive weekly e-mail notifications about fixes and other software support news, follow these steps:

2. Click My support under the Personalized support heading in the upper-right corner of the page.
3. If you have already registered for My support, sign in and skip to the next step. If you have not registered, click register now. Complete the registration form using your e-mail address as your IBM ID and click Submit.

Appendix B. Support information 239
Receiving support updates

4. Click Edit profile.
5. In the Products list, select Software. A second list is displayed.
6. In the second list, select a product segment, for example, Systems Management. A third list is displayed.
7. In the third list, select a product sub-segment, for example, Job Scheduling. A list of applicable products is displayed.
8. Select the products for which you want to receive updates, for example, IBM Tivoli Workload Scheduler and IBM Tivoli Management Framework.
9. Click Add products.
10. After selecting all products that are of interest to you, click Subscribe to email on the Edit profile page.
11. In the Documents list, select Software.
12. Select Please send these documents by weekly email from the list.
13. Update your e-mail address as needed.
14. Select the types of documents that you want to receive information about.
15. Click Update.

If you experience problems with the My support feature, you can obtain help in one of the following ways:

Online
   Send an e-mail message to erchelp@ca.ibm.com, describing your problem.

By phone
   Call 1-800-IBM-4You (1-800-426-4968).
Contacting IBM Software Support

IBM Software Support provides assistance with product defects.

Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have:

- For IBM distributed software products (including, but not limited to, Tivoli, Lotus®, and Rational® products, as well as DB2 and WebSphere products that run on Windows, or UNIX operating systems), enroll in Passport Advantage® in one of the following ways:

  **Online**
  Go to the Passport Advantage Web site at [http://www.lotus.com/services/passport.nsf/WebDocs/Passport_Advantage_Home](http://www.lotus.com/services/passport.nsf/WebDocs/Passport_Advantage_Home) and click How to Enroll.

  **By phone**
  For the phone number to call in your country, go to the IBM Software Support Web site at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region.

- For customers with Subscription and Support (S & S) contracts, go to the Software Service Request Web site at [https://techsupport.services.ibm.com/ssr/login](https://techsupport.services.ibm.com/ssr/login).


- For IBM eServer™ software products (including, but not limited to, DB2 and WebSphere products that run in zSeries, pSeries, and iSeries environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM Technical Support Advantage Web site at [http://www.ibm.com/servers/eserver/techsupport.html](http://www.ibm.com/servers/eserver/techsupport.html).

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States. From other countries, go to the contacts page of the IBM Software Support Handbook on the Web at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region for phone numbers of people who provide support for your location.

To contact IBM Software support, follow these steps:

1. “Determining the business impact” on page 242
2. “Describing problems and gathering information” on page 242
3. “Submitting problems” on page 242
Receiving support updates

Determining the business impact
When you report a problem to IBM, you are asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem that you are reporting. Use the following criteria:

Severity 1
The problem has a critical business impact. You are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.

Severity 2
The problem has a significant business impact. The program is usable, but it is severely limited.

Severity 3
The problem has some business impact. The program is usable, but less significant features (not critical to operations) are unavailable.

Severity 4
The problem has minimal business impact. The problem causes little impact on operations, or a reasonable circumvention to the problem was implemented.

Describing problems and gathering information
When describing a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Software Support specialists can help you solve the problem efficiently. To save time, know the answers to these questions:

- What software versions were you running when the problem occurred?
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
- Can you re-create the problem? If so, what steps were performed to re-create the problem?
- Did you make any changes to the system? For example, did you make changes to the hardware, operating system, networking software, and so on.
- Are you currently using a workaround for the problem? If so, be prepared to explain the workaround when you report the problem.

Submitting problems
You can submit your problem to IBM Software Support in one of two ways:

Online
Click Submit and track problems on the IBM Software Support site at [http://www.ibm.com/software/support/probsub.html](http://www.ibm.com/software/support/probsub.html). Type your information into the appropriate problem submission form.

By phone
For the phone number to call in your country, go to the contacts page of the IBM Software Support Handbook at [http://techsupport.services.ibm.com/guides/contacts.html](http://techsupport.services.ibm.com/guides/contacts.html) and click the name of your geographic region.

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Software Support provides a workaround that you can implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the
Receiving support updates

Software Support Web site daily, so that other users who experience the same problem can benefit from the same resolution.
Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this publication in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user’s responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this publication. The furnishing of this publication does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX  78758  U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this publication and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

For the code entitled Libmsg

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that Alfalfa’s name not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

ALPHALPHA DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL ALPHALPHA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

For the code entitled Btree Library

(C) Copyright, 1988, 1989 Marcus J. Ranum All rights reserved This software, its documentation, and supporting files may be distributed and modified freely subject to the following conditions: 1) None of the copyright headers in any of the files may be altered in any way. 2) This software is not to be incorporated into another software package or operating system that is sold for a fee, or is "shareware", without the author's written consent, unless the source code to this software is included with that package, or a notice is included stating that this software is available without a fee, who wrote it, and where to look for it. 3) This software may not be distributed in binary-only format, unless a notice is included stating that this software is available without a fee, and where to look for it. 4) This software may not be redistributed in any manner that includes a handling charge, without the author's written consent. 5) This software may not be redistributed without its documentation and supporting files. Users are encouraged to modify this software and improve it, and to share those improvements with others. If you feel you have improved it, please contact the author, if you want to share your improvements. Please inform the author of bugs or bug fixes. The author assumes no liability for the use or misuse of this software, and makes no claims as to its suitability for any purpose. Do not take internally. See a physician immediately if accidentally ingested. Marcus J. Ranum 3018 Guilford Ave Baltimore, Md 21218 Dec 19, 1989

For the code entitled ANT

============================================================================
== NOTICE file corresponding to the section 4 of ==
== the Apache License, Version 2.0, ==
== in this case for the Apache Ant distribution. ==
============================================================================
This product includes also software developed by:
- the W3C consortium (http://www.w3c.org),
- the SAX project (http://www.saxproject.org)

Please read the different LICENSE files present in the root directory of this distribution.

END OF ANT NOTICES AND INFORMATION

For the code entitled Regex

* regcomp and regexec -- regsub and regerror are elsewhere
* Copyright (c) 1986 by University of Toronto.
* Written by Henry Spencer. Not derived from licensed software.
* Permission is granted to anyone to use this software for any
* purpose on any computer system, and to redistribute it freely,
* subject to the following restrictions:
* 1. The author is not responsible for the consequences of use of
* this software, no matter how awful, even if they arise
* from defects in it.
* 2. The origin of this software must not be misrepresented, either
* by explicit claim or by omission.
For the code entitled SunRPC

Sun RPC is a product of Sun Microsystems, Inc. and is provided for unrestricted use provided that this legend is included on all tape media and as a part of the software program in whole or part. Users may copy or modify Sun RPC without charge, but are not authorized to license or distribute it to anyone else except as part of a product or program developed by the user.

SUN RPC IS PROVIDED AS IS WITH NO WARRANTIES OF ANY KIND INCLUDING THE WARRANTIES OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR ARISING FROM A COURSE OF DEALING, USAGE OR TRADE PRACTICE.

Sun RPC is provided with no support and without any obligation on the part of Sun Microsystems, Inc. to assist in its use, correction, modification or enhancement.

SUN MICROSYSTEMS, INC. SHALL HAVE NO LIABILITY WITH RESPECT TO THE INFRINGEMENT OF COPYRIGHTS, TRADE SECRETS OR ANY PATENTS BY SUN RPC OR ANY PART THEREOF.

In no event will Sun Microsystems, Inc. be liable for any lost revenue or profits or other special, indirect and consequential damages, even if Sun has been advised of the possibility of such damages.

Sun Microsystems, Inc. 2550 Garcia Avenue Mountain View, California 94043

For System Service Monitors 4.0 Code

Libpcap:

Libpcap 0.5: Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Net SNMP:

University of California. All rights Reserved. Portions copyright 2001-2002 Networks Associates Technology, Inc. All rights reserved. Portions of this code are copyright 2001-2002 Cambridge Broadband Ltd. All rights reserved.

Permission to use, copy, modify and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU and The Regents of the University of California not be used in advertising or publicity pertaining to distribution of the software without specific written permission.

CMU AND THE REGENTS OF THE UNIVERSITY OF CALIFORNIA DISCLAIM ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL CMU OR THE REGENTS OF THE UNIVERSITY OF CALIFORNIA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM THE LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the names of Networks Associates Technology, Inc or Cambridge Broadband Ltd., nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF NET SNMP NOTICES AND INFORMATION
BZIP2:

Bzip2 1.0.2 Copyright 1996-2002 Julian R Seward. All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

END OF BZIP2 NOTICES AND INFORMATION

MD5:

MD5 Message-digest Algorithm April 1992 Copyright 1991-1992 RSA Data Security, Inc. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. D5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software

END OF MD5 NOTICES AND INFORMATION

END OF SYSTEM SERVICE MONITORS NOTICES AND INFORMATION
INSTALLSHIELD MULTI-PLATFORM 11.5 Code

The IBM license agreement and any applicable notices on the web download page for this product refers you to this file for details concerning terms and conditions applicable to software code identified as excluded components in the License Information document and included in InstallShield Multi-Platform 11.5 (the "Program").

Notwithstanding the terms and conditions of any other agreement you may have with IBM or any of its related or affiliated entities (collectively "IBM"), the third party software code identified below are "Excluded Components" and are subject to the terms and conditions of the License Information document accompanying the Program and not the license terms that may be contained in the notices below. The notices are provided for informational purposes.

The Program includes the following Excluded Components:
- Apache Ant V1.5.4
- Jakarta ORO V2.0.8
- Quick V1.0.1
- Xalan V2.5.2
- HSQLDB V1.7.1
- JTOPEN V4.4
- BrowserLauncher V1.4b1
- InfoZip Unzip stub file V5.40, V5.41, V5.42

**APACHE CODE**

The Program includes all or portions of the following software developed by the Apache Software Foundation. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see http://www.apache.org. The following is the list of Excluded Components governed under V1.1 of the Apache license:
- Xalan V2.5.2
- Apache Ant V1.5.4
- Jakarta ORO V2.0.8

IBM is required to provide you, as the recipient of such software, with a copy of the following license from the Apache Software Foundation:

The Apache Software License, Version 1.1

Copyright (c) 2000 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear. 4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org. 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see http://www.apache.org.

Parts of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

END OF APACHE NOTICES AND INFORMATION

**JXML CODE**

The Program is accompanied by the following JXML software:

**Quick V1.0.1**

IBM is required to provide you, as the recipient of such software, with a copy of the following license from JXML:

Copyright (c) 1998, 1999, JXML, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

All product materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by JXML, Inc. and its contributors: http://www.jxml.com/mdsax/contributers.html
Neither name of JXML nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY JXML, INC. AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JXML OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF JXML NOTICES AND INFORMATION.

InfoZip Code
The Program is accompanied by the following InfoZip software:

One or more of: InfoZip Unzipsfx stub file V5.40, V5.41, V5.42

IBM is required to provide you, as the recipient of such software, with a copy of the following license from InfoZip:

This is version 2000-Apr-09 of the Info-ZIP copyright and license. The definitive version of this document should be available at ftp://ftp.info-zip.org/pub/infozip/license.html indefinitely.

Copyright (c) 1990-2000 Info-ZIP. All rights reserved.

For the purposes of this copyright and license, "Info-ZIP" is defined as the following set of individuals:


This software is provided "as is," without warranty of any kind, express or implied. In no event shall Info-ZIP or its contributors be held liable for any direct, indirect, incidental, special or consequential damages arising out of the use of or inability to use this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:
1. Redistributions of source code must retain the above copyright notice, definition, disclaimer, and this list of conditions. 2. Redistributions in binary form must reproduce the above copyright notice, definition, disclaimer, and this list of conditions in documentation and/or other materials provided with the distribution. 3. Altered versions—including, but not limited to, ports to new operating systems, existing ports with new graphical interfaces, and dynamic, shared, or static library versions—must be plainly marked as such and must not be misrepresented as being the original source. Such altered versions also must not be misrepresented as being Info-ZIP releases—including, but not limited to, labeling of the altered versions with the names “Info-ZIP” (or any variation thereof, including, but not limited to, different capitalizations), “Pocket UnZip,” “WiZ” or “MacZip” without the explicit permission of Info-ZIP. Such altered versions are further prohibited from misrepresentative use of the Zip-Bugs or Info-ZIP e-mail addresses or of the Info-ZIP URL(s). 4. Info-ZIP retains the right to use the names “Info-ZIP,” “Zip,” “UnZip,” “WiZ,” “Pocket UnZip,” “Pocket Zip,” and “MacZip” for its own source and binary releases.

END OF INFOZIP NOTICES AND INFORMATION

HSQL CODE

The Program is accompanied by the following HSQL Development Group software:

HSQldb V1.7.1

IBM is required to provide you, as the recipient of such software, with a copy of the following license from the HSQL Development Group:

Copyright (c) 2001-2002, The HSQL Development Group All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the HSQL Development Group nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL HSQL DEVELOPMENT GROUP, HSQldb.ORG, OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF HSQL NOTICES AND INFORMATION

**BROWSERLAUNCHER code**

The Program is accompanied by the following BrowserLauncher software:

BrowserLauncher V1.4b1

IBM is required to provide you, as the recipient of such software, with a copy of the following license from BrowserLauncher:

This code is Copyright 1999-2001 by Eric Albert (ejalbert@cs.stanford.edu) and may be redistributed or modified in any form without restrictions as long as the portion of this comment from this paragraph through the end of the comment is not removed. The author requests that he be notified of any application, applet, or other binary that makes use of this code, but that’s more out of curiosity than anything and is not required. This software includes no warranty. The author is not responsible for any loss of data or functionality or any adverse or unexpected effects of using this software.

Credits: Steven Spencer, JavaWorld™ magazine (http://www.javaworld.com/javaworld/javatips/jw-javatip66.html) Thanks also to Ron B. Yeh, Eric Shapiro, Ben Engber, Paul Teitlebaum, Andrea Cantatore, Larry Barowski, Trevor Bedzek, Frank Miedrich, and Ron Rabakkk

@author Eric Albert (ejalbert@cs.stanford.edu) @version 1.4b1 (Released June 20, 2001)

END OF BROWSERLAUNCHER NOTICES AND INFORMATION

**JTOPEN Code**

The Program is accompanied by the following JTOPEN software:

JTOPEN V4.4

The source code for JTOPEN is available on the Web from: http://www-03.ibm.com/servers/eserver/iseries/toolbox/downloads.htm

END OF JTOPEN NOTICES AND INFORMATION

END OF INSTALLSHIELD MULTI-PLATFORM NOTICES AND INFORMATION

---

For the code entitled embedded version of IBM WebSphere Application Server ("embedded WebSphere Application Server") V6.1

NOTICES AND INFORMATION FOR EXCLUDED COMPONENTS

The IBM license agreement and any applicable information on the web download page for IBM products refers You to this file for details concerning notices applicable to software code identified as Excluded Components in the License
Information document and included in the IBM WEBSPHERE APPLICATION SERVER product listed above (“the Program”).

Notwithstanding the terms and conditions of any other agreement you may have with IBM or any of its related or affiliated entities (collectively “IBM”), the third party software code identified below are “Excluded Components” and are subject to the terms and conditions of the License Information document accompanying the Program and not the license terms that may be contained in the notices below. The notices are provided for informational purposes.

Please note: This Notices file may identify information or Excluded Components listed in the agreements for the Program that are not used by, or that were not shipped with, the Program as you installed it. The Program includes the following Excluded Components:

* Apache Ant v. 1.6.5
* Apache Commons-Discovery v. 0.2
* Apache Commons-Logging v. 1.0.3
* Apache Derby v. 10.1
* Apache Jakarta JSP Standard Tag Library (JSTL) Code v. 1.1
* Apache Jakarta Struts 1.1
* Apache Soap v. 2.3.1
* Apache Web Services Invocation Framework Code (WSIF) v. 2.0
* Cup Parser Generator for Java 1.0
* EMF 2.1.1
* International Components for Unicode (ICU) ICU4J version 3.4
* JYTHON 2.1.3
* OSGi Materials
* XML4C v 5.5.0
* XML SCHEMAS

**Apache code**

APACHE CODE. The Program includes all or portions of the following software which is subject to the Apache License Version 2.0, January 2004. This software may also consist of voluntary contributions made by many individuals to the Apache Software Foundation. For more information on the Apache Software Foundation, please see http://www.apache.org:

* Apache Ant 1.6.5
* Apache Commons-Discovery v. 0.2
* Apache Commons-Logging v. 1.0.3
* Apache Derby v. 10.1
* Apache Jakarta JSP Standard Tag Library (JSTL) Code v. 1.1
* Apache Jakarta Struts v.1.1
* Apache Soap v. 2.3.1
* Apache Web Services Invocation Framework Code (WSIF) v. 2.0

IBM is required to provide you, as the recipient of such software, with a copy of the following license notice from the Apache Software Foundation:

Apache License Version
2.0, January 2004
http://www.apache.org/licenses/

**TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION**

1. Definitions.

License” shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

You” (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below.

"Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the
Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF APACHE NOTICES AND INFORMATION

CUP Parser Generator for Java
The Program includes CUP Parser Generator ("JCUP") version 1.0 software which is Copyright 1996-1999 by Scott Hudson, Frank Flannery and C. Scott Ananian. All rights reserved. IBM obtained the JCUP Software under the terms and conditions of the following license:

CUP Parser Generator Copyright Notice, License, and Disclaimer

Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The open source code (Cup) is provided as is, without warranties of any kind, including warranties of noninfringement, merchantability and fitness for a particular purpose. IBM will not be responsible for any damages resulting directly from the use of this code, except to the extent that IBM may, at its option, debug or replace the defective code.
The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

END OF CUP PARSER GENERATOR FOR JAVA NOTICES AND INFORMATION

EPL CODE

The Program includes all or portions of the following software which IBM obtained under the terms and conditions of the Common Public License ("CPL Code") or the Eclipse Public License ("EPL Code").

Eclipse EMF 2.1.2, and OSGI Materials You may download the source version of this component at: http://www.eclipse.org/

END OF EPL CODE NOTICES AND INFORMATION

JYTHON.

The Program includes Jython 2.1.3 software which is Copyright (c) 2000, Jython Developers. All rights reserved. IBM did not modify the Jython software and obtained it under the terms and conditions of the following license:

Copyright (c) 2000, Jython Developers All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Jython Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF JYTHON NOTICES AND INFORMATION
XML SCHEMAS

The Program includes XML SCHEMAS which were obtained by IBM under the following terms and conditions:

W3C(R) SOFTWARE NOTICE AND LICENSE


This W3C work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code: "Copyright (C) [date-of-software] World Wide Web Consortium, Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal

3. Notice of any changes or modifications to the W3C files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.
The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

END OF XML SCHEMA NOTICES

**ICU v 3.4 International Components for Unicode (ICU):**

International Components for Unicode (ICU) : The program includes ICU4J version 3.4 The following terms and conditions apply to ICU code : Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners

END OF ICU NOTICES

**OASIS XML SCHEMAS**

The Program includes XML SCHEMAS which were obtained by IBM under the following terms and conditions:

Copyright (C) OASIS Open (2004-2005). All Rights Reserved

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights
defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

END OF OASIS XML SCHEMA NOTICES

END OF THE WEBSPHERE APPLICATION SERVER NOTICES AND INFORMATION.

For the code entitled Time Zone Library

Copyright (c) 1985, 1987, 1988 The Regents of the University of California.

All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

For the code entitled SNMP Library

Copyright 1988, 1989 by Carnegie Mellon University

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of CMU not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

CMU DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL CMU BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE

Notices 263
For the code entitled TCL

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF; EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

For the code entitled Test Library

Copyright (c) 1992, 1993, 1994

The Regents of the University of California. All rights reserved.

This code is derived from software contributed to Berkeley by Kenneth Almquist.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. 3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the University of California, Berkeley and its contributors. 4. Neither the name of the
University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

For the code entitled ICU4C, ICU4J and ICU4JNI:

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2005 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE
Unicode Data Files include all data files under the directories:

http://www.unicode.org/Public/ and http://www.unicode.org/reports/.

Unicode Software includes any source code under the directories

http://www.unicode.org/Public/ and http://www.unicode.org/reports/.

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1991-2004 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and associated documentation (the "Data Files") or Unicode software and associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear in all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

For the code entitled zLib

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.
Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

For the code entitled OpenSSL

OpenSSL License

LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License
/*                                  *                        * Copyright (c) 1998-2003 The OpenSSL Project. All rights reserved. *                                  * Redistribution and use in source and binary forms, with or without * modification, are permitted provided that the following conditions * are met: *                                  * 1. Redistributions of source code must retain the above copyright  * notice, this list of conditions and the following disclaimer. * 2. Redistributions in binary form must reproduce the above copyright * notice, this list of conditions and the following disclaimer in * the documentation and/or other materials provided with the * distribution. * 3. All advertising materials mentioning features or use of this * software must display the following acknowledgment: * "This product includes software developed by the OpenSSL Project * for use in the OpenSSL Toolkit. (http://www.openssl.org)/" * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to * endorse or promote products derived from this software without * prior written permission. For written permission, please contact * openssl-core@openssl.org. * 5. Products derived from this software may not be called "OpenSSL" * nor may "OpenSSL" appear in their names without prior written * permission of the OpenSSL Project. * 6. Redistributions of any form whatsoever must retain the following * acknowledgment: * "This product includes software developed by the OpenSSL Project * for use in the OpenSSL Toolkit (http://www.openssl.org)/" * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT `"AS IS' AND ANY * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.

* This product includes cryptographic software written by Eric Young
  (eay@cryptsoft.com). This product includes software written by Tim
  Hudson (tjh@cryptsoft.com).

*/

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * Ihash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *       "This product includes cryptographic software written by
 *        Eric Young (eay@cryptsoft.com)"
 *    The word 'cryptographic' can be left out if the rouines from the library
 *    being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 *    the apps directory (application code) you must include an acknowledgement:
 *    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

For the code entitled IBM DB2 Version 9

NOTICES AND INFORMATION

The IBM license agreement and any applicable information on the web download page for IBM products refers You to this file for details concerning notices applicable to software code identified as Excluded Components in the License Information document and included in the IBM DB2 product listed above ("the Program").

Notwithstanding the terms and conditions of any other agreement you may have with IBM or any of its related or affiliated entities (collectively "IBM"), the third party software code identified below are “Excluded Components” and are subject to the terms and conditions of the License Information document accompanying the Program and not the license terms that may be contained in the notices below. The notices are provided for informational purposes.

The Program includes the following Excluded Components:

* Expat 1.95.1
* CUP Parser Generator 0.10k
* Apache Common-FileUpload 1.0
* Apache Common-Validator 1.0.2
* Apache Common-Collections 2.1
* Apache Common-Digestor 1.5
* APache Common-BeanUtils 1.6
* Apache Jakarta Struts 1.1
* Apache Tomcat 4.1.30
* Apache Lucene 1.4.3
* Apache Lucene HTML Parser 1.4.3
* Apache Ant v. 1.6.5
* Apache Commons-Discovery v. 0.2
* Apache Commons-Logging v. 1.0.3
* Apache Derby v. 10.1
* Apache Jakarta JSP Standard Tag Library (JSTL) Code v. 1.1
* Apache Jakarta Struts 1.1
* Apache Soap v. 2.3.1
* Apache Web Services Invocation Framework Code (WSIF) v. 2.0
* XMLAC v 5.5.0/5.6
* XSLTAC 1.10
* Eclipse SDK 3.1.2
* Eclipse Help System 3.1.1
* Eclipse Modeling Framework 1.1.1/2.1.1
* Eclipse Web Tools Platform (WTP) 1.0.2
* OSGi Materials
* GEF 3.1.1
* XSD 2.1.2
* TPTP 4.0.1.1
* UML2 1.1.1
* ICU4C 3.2.1
EXPAT CODE

The Program includes Expat 1.95.1 software which is Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd. IBM obtained the Expat Software under the terms and conditions of the following license:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the"Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

END OF EXPAT NOTICES AND INFORMATION

CUP PARSER GENERATOR FOR JAVA

The Program includes CUP Parser Generator ("JCUP") version 1.0 software which is Copyright 1996-1999 by Scott Hudson, Frank Flannery and C. Scott Ananian. All rights reserved. IBM obtained the JCUP Software under the terms and conditions of the following license:

CUP Parser Generator Copyright Notice, License, and Disclaimer

Copyright 1996-1999 by Scott Hudson, Frank Flannery, C. Scott Ananian

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data
or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

END OF CUP PARSER GENERATOR FOR JAVA NOTICES AND INFORMATION

**APACHE CODE**

The Program includes all or portions of the following software developed by the Apache Software Foundation. IBM obtained the Apache Code under the terms and conditions of the Apache License. This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see [ftp://www.apache.org](ftp://www.apache.org):

* Apache Common-FileUpload 1.0
* Apache Common-Validator 1.0.2
* Apache Common-Collections 2.1
* Apache Common-Digestor 1.5
* APACHE Common-BeanUtils 1.6
* Apache Jakarta Struts 1.1
* Apache Tomcat 4.1.30
* Apache Lucene 1.4.3
* Apache Lucene HTML Parser 1.4.3
* Apache Ant 1.6.5
* Apache Commons-Discovery v. 0.2
* Apache Commons-Logging v. 1.0.3
* Apache Derby v. 10.1
* Apache Jakarta JSP Standard Tag Library (JSTL) Code v. 1.1
* Apache Jakarta Struts v.1.1
* Apache Soap v. 2.3.1
* Apache Web Services Invocation Framework Code (WSIF) v. 2.0
* XSLT4C 5.6
* XSLT4C 1.10

IBM is required to provide you, as the recipient of such software, with a copy of the following license(s) from the Apache Software Foundation:

```*/
* The Apache Software License, Version 1.1
* Copyright (c) 2000 The Apache Software Foundation. All rights reserved.
* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
* "This product includes software developed by the Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
* 4. The names "Apache" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
*/```
5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see http://www.apache.org/

 Portions of this software are based upon public domain software originally written at the National Center for Supercomputing Applications, University of Illinois, Urbana-Champaign.

Apache LicenseVersion 2.0, January 2004 http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions. "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
“Work” shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

“Derivative Works” shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

“Contribution” shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, “submitted” means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as “Not a Contribution.”

“Contributor” shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and (b) You must cause any modified files to carry prominent notices stating that You changed the files; and (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work, excluding those notices that
do not pertain to any part of the Derivative Works; and (d) If the Work includes a
"NOTICE" text file as part of its distribution, then any Derivative Works that You
distribute must include a readable copy of the attribution notices contained within
such NOTICE file, excluding those notices that do not pertain to any part of the
Derivative Works, in at least one of the following places: within a NOTICE text file
distributed as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or, within a display
generated by the Derivative Works, if and wherever such third-party notices
normally appear.

The contents of the NOTICE file are for informational purposes only and do not
modify the License. You may add Your own attribution notices within Derivative
Works that You distribute, alongside or as an addendum to the NOTICE text from
the Work, provided that such additional attribution notices cannot be construed as
modifying the License.

You may add Your own copyright statement to Your modifications and may
provide additional or different license terms and conditions for use, reproduction,
or distribution of Your modifications, or for any such Derivative Works as a whole,
provided Your use, reproduction, and distribution of the Work otherwise complies
with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any
Contribution intentionally submitted for inclusion in the Work by You to the
Licensor shall be under the terms and conditions of this License, without any
additional terms or conditions. Notwithstanding the above, nothing herein shall
supersede or modify the terms of any separate license agreement you may have
executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names,
trademarks, service marks, or product names of the Licensor, except as required for
reasonable and customary use in describing the origin of the Work and
reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in
writing, Licensor provides the Work (and each Contributor provides its
Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS
OF ANY KIND, either express or implied, including, without limitation, any
warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY,
or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for
determining the appropriateness of using or redistributing the Work and assume
any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort
(including negligence), contract, or otherwise, unless required by applicable law
(such as deliberate and grossly negligent acts) or agreed to in writing, shall any
Contributor be liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a result of this
License or out of the use or inability to use the Work (including but not limited to
damages for loss of goodwill, work stoppage, computer failure or malfunction, or
any and all other commercial damages or losses), even if such Contributor has
been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or
Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance
of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF APACHE NOTICES AND INFORMATION

EPL CODE

The Program includes all or portions of the following software which IBM obtained under the terms and conditions of the Common Public License ("CPL Code") or the Eclipse Public License ("EPL Code").

* Eclipse SDK 3.1.2
* Eclipse Help System 3.1.1
* Eclipse Modeling Framework 1.1.1/2.1.2
* Eclipse Web Tools Platform 1.0.2,
* OSGi Materials
* GEF 3.1.1
* XSD 2.1.2
* TPTP 4.0.1.1
* UML2 1.1.1

You may download the source version of these components at:

END OF EPL CODE NOTICES AND INFORMATION

ICU:

The Program includes all or portions of the following software which IBM obtained under the terms and conditions of the following license(s):

* ICU4C 3.2.1
* ICU4J 3.4.3

ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2005 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE
FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories: http://www.unicode.org/Public/ and http://www.unicode.org/reports/. Unicode Software includes any source code under the directories http://www.unicode.org/Public/ and http://www.unicode.org/reports/.

NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.’S DATA FILES (“DATA FILES”), AND/OR SOFTWARE (“SOFTWARE”), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1991-2004 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and associated documentation (the "Data Files") or Unicode software and associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear in all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

END OF ICU NOTICES AND INFORMATION

MX4J:

The Program includes MX4J software and IBM obtained the MX4J Software under the terms and conditions of the following license:

The MX4J License, Version 1.0 Copyright (c) 2001 MX4J. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment: "This product includes software developed by the MX4J project." Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "MX4J" and "mx4j" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact Simone Bordet or Carlos Quiroz.

5. Products derived from this software may not be called "MX4J", nor may "MX4J" appear in their name, without prior written permission of Simone Bordet.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CHRIS SEGUIN OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF MX4J NOTICES AND INFORMATION
UnZipSFX Stub file:

The Program includes "UnZipSFX stub" software from the Info-ZIP group under the terms and conditions of the following license: http://www.info-zip.org/license.html

Copyright (c) 1990-2003 Info-ZIP. All rights reserved.

For the purposes of this copyright and license, "Info-ZIP" is defined as the following set of individuals:


This software is provided "as is," without warranty of any kind, express or implied. In no event shall Info-ZIP or its contributors be held liable for any direct, indirect, incidental, special or consequential damages arising out of the use of or inability to use this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. Redistributions of source code must retain the above copyright notice, definition, disclaimer, and this list of conditions.

2. Redistributions in binary form (compiled executables) must reproduce the above copyright notice, definition, disclaimer, and this list of conditions in documentation and/or other materials provided with the distribution. The sole exception to this condition is redistribution of a standard UnZipSFX binary (including SFXWiz) as part of a self-extracting archive; that is permitted without inclusion of this license, as long as the normal SFX banner has not been removed from the binary or disabled.

3. Altered versions--including, but not limited to, ports to new operating systems, existing ports with new graphical interfaces, and dynamic, shared, or static library versions--must be plainly marked as such and must not be misrepresented as being the original source. Such altered versions also must not be misrepresented as being Info-ZIP releases--including, but not limited to, labeling of the altered versions with the names "Info-ZIP" (or any variation thereof, including, but not limited to, different capitalizations), "Pocket UnZip," "WiZ" or "MacZip" without the explicit permission of Info-ZIP. Such altered versions are further prohibited from misrepresentative use of the Zip-Bugs or Info-ZIP e-mail addresses or of the Info-ZIP URL(s).


END OF UNZIP SFX NOTICES AND INFORMATION

ZLIB

The Program includes the zlib.h - interface of the 'zlib' general purposes compression library version 1.2.2. IBM obtained the software under the terms and conditions of the following license agreement:

http://www.gzip.org/zlib/zlib_license.html
License

Copyright (C) 1995-2002 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org
Mark Adler madler@alumni.caltech.edu

END OF ZLIB NOTICES AND INFORMATION

RSA BSAFE Crypto-C Software

Copyright RSA 2002.

END OF CRYPTO-C SOFTWARE NOTICES AND INFORMATION

MINIZIP

The Program includes the zip.h - IO for compress .zip files version 1.01. IBM obtained the software under the terms and conditions of the following license agreement:

Copyright (C) 1998-2004 Gilles Vollant

This software is provided 'as-is', without any express or implied warranty. In no event will the author be liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

END OF MINIZIP NOTICES AND INFORMATION

JYTHON

The Program includes Jython 2.1.3 software which is Copyright (c) 2000, Jython Developers. All rights reserved. IBM did not modify the Jython software and obtained it under the terms and conditions of the following license:

Copyright (c) 2000, Jython Developers All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Jython Developers nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

END OF JYTHON NOTICES AND INFORMATION

XML SCHEMAS:
The Program includes XML SCHEMAS which were obtained by IBM under the following terms and conditions:

W3C(R) SOFTWARE NOTICE AND LICENSE


This W3C work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code: "Copyright (C) [date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/"

3. Notice of any changes or modifications to the W3C files, including
the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

END OF XML SCHEMA NOTICES

OASIS XML SCHEMAS:

The Program includes XML SCHEMAS which were obtained by IBM under the following terms and conditions:

Copyright (C) OASIS Open (2004-2005). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to OASIS, except as needed for the purpose of developing OASIS specifications, in which case the procedures for copyrights defined in the OASIS Intellectual Property Rights document must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

END OF OASIS XML SCHEMA NOTICES

END OF DB2 V9 NOTICES

---

**Trademarks**

IBM, the IBM logo,i5/OS®, BookManager, DB2 Universal Database, developerWorks, eServer, IBMLink, iSeries, LoadLeveler, Lotus, Maestro, NetView, Passport Advantage, pSeries, Rational, Redbooks, s/390, Tivoli, the Tivoli logo, Tivoli Enterprise Console, WebSphere, z/OS, and zSeries are trademarks of International Business Machines Corporation in the United States, other countries, or both.
Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.
Glossary

A

access method. An executable file used by extended agents to connect to and control jobs on other operating systems (for example, z/OS) and applications (for example, Oracle Applications, PeopleSoft, and SAP R/3). The access method is specified in the workstation definition for the extended agent. See also extended agent.

actual start time. The time that a Tivoli Workload Scheduler job instance or job stream instance actually starts. See also:
- earliest start time
- latest start time
- planned start time
- scheduled time

ad hoc job. A job that is inserted into the current production plan. These jobs are unique to the plan, and are not saved in the database. See also:
- database
- plan

ad hoc prompt dependency. A prompt dependency that is defined within the properties of a job or job stream and is unique to that job or job stream. See also prompt dependency.

agent. An installed component that enables jobs to be run on a computer or a computer partition, provided that the computer or computer partition is also defined as a workstation in the Tivoli Workload Scheduler database. Agents can be standard, fault-tolerant, extended, or network. Specially configured agents are also used as backups for domain managers and the master domain manager. See also:
- backup domain manager
- backup master domain manager
- fault-tolerant agent
- network agent
- standard agent
- extended agent

audit. A process that logs modifications to the database and plan.

B

batchman. A production control process that interacts directly with a copy of the Symphony file distributed to workstations at the beginning of the production period and updates it, resolving dependencies. It is the only process that can update the Symphony file. See also:
- processes
- production period
- symphony file

backup domain manager. An agent in a distributed Tivoli Workload Scheduler network that can assume the responsibilities of its domain manager. It is installed as a full status, fault-tolerant agent. See also:
- fault-tolerant agent
- full status
- domain manager

backup master domain manager. An agent in a distributed Tivoli Workload Scheduler network that can assume the responsibilities of the master domain manager. It is installed as a full status, fault-tolerant agent. See also:
- fault-tolerant agent
- full status
- master domain manager

C

calendar. A list of scheduling dates. Calendars are defined in the database and are mostly assigned to run cycles. Calendars can be used either to identify the dates when job streams or jobs can be run (when used with inclusive run cycles), or when they cannot be run (when used with exclusive run cycles). A calendar can also be designated for use as a freedays calendar in a job stream. See also:
- exclusive run cycle
- freedays calendar
- inclusive run cycle

carry forward. If a job stream is not completed before the end of the current production period it can be carried forward to the next and then to subsequent periods, until the latest start time is reached or the job completes. See also latest start time.

command-line client. A component you use to run selected Tivoli Workload Scheduler master domain manager commands from any workstation where it is installed. It communicates by TCP/IP with the command-line server, which is part of the master domain manager. The command-line client does not need to be installed on the master domain manager, and is a selectable option for installation on other nodes in the network. For details of the supported
command-line server • engine

commands see the Tivoli Workload Scheduler: Planning and Installation Guide. See also master domain manager.

command-line server. See command line client.

conman. A command-line program for monitoring and managing the production environment. See also processes.

composer. A command-line program for managing the definitions of scheduling objects in the database. See also database.

connector. An installed component that provides the interface between the Job Scheduling Console and the engine. See also:

- engine
- job scheduling console

CPU. See workstation.

cpu time. The processor time used by a job. See also duration.

D

database. Contains definitions for scheduling objects (such as jobs, job streams, resources, and workstations). The database also contains data such as job and job stream statistics, user data, and the last time an object was modified. See also plan.

deadline. The time by which a job or job stream is set to complete. When a job or job stream passes the deadline, notifications are sent to users and integrated applications, but the job or job stream is not prevented from running if all time restrictions and dependencies are satisfied. Jobs or job streams that have not yet started or that are still running after the deadline time has expired are considered "late" in the plan. See also plan.

dependency. A prerequisite that must be satisfied before a job or job stream can start. See also:

- external dependency
- file dependency
- follows dependency
- prompt dependency
- resource dependency

distributed network. A connected group of workstations that use the Tivoli Workload Scheduler distributed engine to perform workload scheduling. See also:

- engine
- workstation

distributed workstation. A workstation on which jobs and job streams are run using the distributed engine. See also:

- engine
- workstation

domain. A named group of workstations in a distributed Tivoli Workload Scheduler network, consisting of one or more agents and a domain manager acting as the management hub. All domains have a parent domain except for the master domain. See also:

- domain manager
- master domain manager

domain manager. An installed component in a distributed Tivoli Workload Scheduler network that is the management hub in a domain. All communication to and from the agents in the domain is routed through the domain manager. See also workstation.

duration. The elapsed time that a job is expected to take to complete (estimated duration) and actually takes (actual duration). See also:

- cpu time
- time restriction

E

earliest start time. The time before which a job or job stream cannot start. The job or job stream can start after the earliest start time provided that all other time restrictions and dependencies are satisfied. It is set using the at Job Scheduling Console option or in the command-line scheduling language using the at keyword. See also:

- actual start time
- latest start time
- planned start time
- scheduled time

derived network. A network obtained by connecting one or more Tivoli Workload Scheduler fault-tolerant agents in a distributed network to a Tivoli Workload Scheduler for z/OS node in a z/OS network using TCP/IP, to perform workload scheduling. In this configuration, the Tivoli Workload Scheduler for z/OS node becomes the master domain manager of the fault-tolerant agents to schedule and manage jobs in the distributed network. See also:

- engine
- workstation

engine. The core software for the scheduling environment. The engine can be either a z/OS engine (installed as part of the product "Tivoli Workload Scheduler for z/OS") or a distributed engine (installed as part of the product "Tivoli Workload Scheduler").
exclusive run cycle. A run cycle that specifies the days and times that a job stream cannot be run. Exclusive run cycles take precedence over inclusive run cycles. See also [run cycle].

explorer view. A graphical view in the Job Scheduling Console used to modify and maintain job streams in the database and the plan. See also:

- [database]
- [plan]
- [views]

extended agent. An agent used to integrate Tivoli Workload Scheduler job control features with other operating systems (for example, z/OS) and applications (for example, Oracle Applications, PeopleSoft, and SAP R/3). Extended agents must be hosted by a master domain manager, domain manager, or an agent (not another extended agent), and use access methods to communicate with external systems. See also [access method].

external dependency. A dependency defined in one job or job stream that refers to another job stream or to a job in another job stream.

external job. A job referred to in an external dependency. See also [external dependency].

F

fault-tolerant agent. A installed agent component in a distributed Tivoli Workload Scheduler network capable of resolving local dependencies and launching its jobs in the absence of a domain manager.

fence. Regulates whether a job can be run on a workstation. The job fence is a priority level that the priority of a job must exceed before it can run.

file dependency. A dependency where a job or job stream cannot start until it finds a specific file is present in a specific path on a specific workstation. Sometimes called an opens file dependency. See also [dependency].

final job stream. The last job stream that is run in a production period. It contains scripts that generate the next production plan. See also:

- [production period]
- [production plan]

follows dependency. A dependency where a job or job stream cannot start until other jobs or job streams have completed successfully. See also [dependency].

forecast plan. A projection over a selected timeframe based on the job streams and dependencies defined in the database. See also:

- [database]

exclusive run cycle • iCalendar

plan

freedays calendar. A calendar assigned to a job stream to represent the non-working days when job streams and jobs are not to be run. It can also be used to designate Saturdays or Sundays, or both, as workdays. See also:

- [calendar]
- [holidays calendar]

FTA. See [fault-tolerant agent].

full status. An attribute of an agent that enables it to be updated with the status of jobs and job streams running on all other workstations in its domain and in subordinate domains, but not on peer or parent domains. A backup domain manager or master domain manager must be full status. See also:

- [backup domain manager]
- [domain]
- [master domain manager]

G

global options. Configuration options defined on the master domain manager using optman. These options apply to all workstations in the Tivoli Workload Scheduler network. See also:

- [local options]
- [optman]
- [user options]

graph view. A graphical view in the Job Scheduling Console used to modify and maintain job streams in the database and the plan. See also:

- [database]
- [plan]
- [views]

H

holidays calendar. The default freedays calendar for all job streams. It is called "holidays". See also:

- [calendar]
- [freedays calendar]

host. A workstation required by extended agents. It can be any Tivoli Workload Scheduler workstation except another extended agent.

I

iCalendar. A standard (RFC 2445) for calendar data exchange. Specific iCalendars can be supplied in place of Tivoli Workload Scheduler calendars to determine the dates on which jobs or job streams should run. See also [calendar].
impact view • local options

impact view.  A graphical view in the Job Scheduling Console used to modify and maintain job stream instance dependencies in the plan. See also:
• plan
• views

inclusive run cycle.  A run cycle that specifies the days and times that a job stream is scheduled to be run. Exclusive run cycles take precedence over inclusive run cycles. See also run cycle.

interactive jobs.  A job that runs interactively on a Windows desktop.

internal status.  The current status of jobs and job streams in the Tivoli Workload Scheduler engine. The internal status is unique to Tivoli Workload Scheduler. See also status.

internetwork dependencies.  A dependency between jobs or job streams in separate Tivoli Workload Scheduler networks. See also network agent.

internetwork job or job stream.  A job or job stream in a remote Tivoli Workload Scheduler network that is referenced by an internetwork dependency defined for a job or job stream in the local network. See also network agent.

J

Jnextday.  The previously used term for: JnextPlan.

Jnextplan.  A job that creates or extends the production plan. See also production plan.

job.  A unit of work that is processed at a workstation. The job definition consists of a unique job name in the database along with other information necessary to run the job. See also job definition.

job definition.  A definition of a unit of work that resides in the database of the distributed Tivoli Workload Scheduler engine and can be added to a job stream. Job definitions can be created before creating a job stream, or can be created as part of the creation or modification of a job stream. See also job stream.

job instance.  A job scheduled for a specific run date in the plan. See also job.

job scheduling console.  A Java graphical user interface used to create, modify, and maintain job scheduling objects, and to manage the production environment. See also views on page 289.

job limit.  See limit.

job status.  See status.

job stream.  A list of jobs that run as a unit (such as a weekly backup application), along with run cycles, times, priorities, and other dependencies that determine the exact order in which the jobs run.

job stream instance.  A job stream that is scheduled for a specific run date in the plan. See also job stream.

jobman.  A job management process that controls the launching of jobs under the direction of batchman and reports job status back to mailman. The jobman process is responsible for tracking job states and for setting the environment as defined in .jobmanrc when requesting job launches. See also:
• batchman
• jobman
• mailman

jobmon.  A job management and monitoring process in the Windows version of Tivoli Workload Scheduler. A separate jobmon process is spawned to launch and monitor each job. It reports job status back to jobman. See also:
• jobman
• processes

JSC.  See Job scheduling console.

L

latest start time.  The time before which the job or job stream must start. The job or job stream can start before the latest start time provided that all other dependencies are satisfied. It is set in the command-line scheduling language using the until keyword. See also:
• actual start time
• earliest start time
• planned start time
• scheduled time

limit.  A means of allocating a specific number of job slots into which Tivoli Workload Scheduler is allowed to launch jobs. A job limit can be set for each job stream, and for each workstation. For example, setting the workstation job limit to 25 permits Tivoli Workload Scheduler to have no more than 25 jobs running concurrently on the workstation.

list.  A means of filtering plan and database objects and presenting them in a table.

local options.  Configuration options defined on each workstation in the localopts file. Each workstation in the Tivoli Workload Scheduler network must have a localopts file. The settings in this file are changed using a text editor, and apply only to that workstation. See also:
• global options
• user options
logman. A command that produces job statistics from the previous production plan log file, and updates the preproduction plan.

M

makesec. A command-line utility that compiles the security file. See also security file.

mailman. A mail management process. It routes messages to local and remote workstations. Additional mailman processes named ServerIDs are created on domain managers to divide the load on mailman and improve the efficiency of message handling. When the domain manager starts up, it creates a separate mailman process instance for each ServerID specified in the workstation definitions of the agents it manages. Each workstation then contacts its own ServerID on the domain manager instead of contacting the main mailman process. See also processes.

master domain manager. An installed component that performs the role of management hub of the top-level domain in the Tivoli Workload Scheduler network. It maintains the database of all scheduling objects in the domain and the central configuration files. The master domain manager generates the plan and creates and distributes the Symphony file. In addition, logs and reports for the network are maintained on the master domain manager. See also:

- backup master domain manager
- database
- domain
- plan

MDM. See master domain manager.

metronome. An application that takes a snapshot of the Tivoli Workload Scheduler configuration and generates an HTML report. It is used in problem determination to provide information to IBM Software Support.

mozart. The previously used term for the database.

N

netman. A network management process that is started by the Startup script in UNIX, or as a service in Windows. Netman behaves like a network listener program which receives conman start, stop, link or unlink requests from the network. The netman process examines each request received and either implements the request itself or spawns a local Tivoli Workload Scheduler process to do so. See also processes.

network agent. A logical extended agent used to create dependencies between jobs and job streams on separate Tivoli Workload Scheduler networks. See also internetwork dependencies.

logman • predefined prompt dependency

O

offset-based run cycle. A run cycle that uses a combination of user-defined periods and offsets. For example, an offset of 3 in a period of 15 days is the third day from the beginning of the period. It is more practical to use offset-based run cycles when the cycle is based on cyclic periods. This term is only used as such in Tivoli Workload Scheduler for z/OS, but the concept applies also to the distributed product. See also:

- rule-based run cycle
- run cycle

opens file dependency. See file dependency.

optman. A command-line program that maintains the global options in the product database.

P

parameter. An entity that enables job instance-specific values to be substituted in job and job stream scripts, either from values in the database or at run time. Parameters cannot be used when scripting extended agent jobs.

plan. The means of scheduling jobs. Objects in the database become instances in the plan. See also:

- database
- final job stream
- forecast plan
- inextPlan
- plan
- preproduction plan
- production plan
- trial plan

planman. An application you use to create, extend, and reset plans of all types. See also plan.

planned start time. The time that Tivoli Workload Scheduler estimates a job instance will start. This estimate is based on start times of previous instances of the job. See also:

- actual start time
- earliest start time
- latest start time
- scheduled time

predecessor. A job or job stream that must complete successfully before successor jobs or job streams can be started. See also successor.

predefined prompt dependency. A prompt dependency that is defined in the database and can be associated to any job or job stream. See also prompt dependency.
priority. A way of determining the order in which jobs and job streams start. Priorities for each job and job stream range from 0 to 101. A job or job stream with a priority of 0 does not run.

preproduction plan. A high-level plan of system activity containing job streams and dependencies. It is created automatically when the production plan is created for the first time. It is extended if the production plan is extended. It is similar to the long-term plan used in Tivoli Workload Scheduler for z/OS. See also [plan].

production period. The time frame covered by the production plan. See also [production plan].

production plan. Contains all job scheduling activity planned for a period. The plan is created or extended by the Jnextplan job or by planman. It is stored in the Symphony file, and consists of all the jobs, job streams, and dependency objects that are scheduled to run for that period, including any jobs or job streams carried forward from the previous plan. See also:
- carry forward
- JnextPlan
- plan

processes. Network processes that control the production environment and network traffic. See also:
- batchman
- jobman
- jobmon
- mailman
- netman
- writer

prompt dependency. A dependency where an operator must respond affirmatively to a prompt so that the dependent job or job stream can run. See also:
- ad hoc prompt dependency
- predefined prompt dependency

R

resource. Either physical or logical system resources. Resources are used as dependencies for jobs and job streams. See also [resource dependency].

resource dependency. A dependency where a job or job stream cannot start until the required quantity of the defined resource is available. See also [resource].

rule-based run cycle. A run cycle that uses rules based on lists of ordinal numbers, types of days, and common calendar intervals (or period names in Tivoli Workload Scheduler for z/OS). For example, the last Thursday of every month. Rule-based run cycles are based on conventional periods, such as calendar months, weeks of the year, and days of the week. In Tivoli Workload Scheduler for z/OS, run cycles can also be based on periods that you define, such as a semester. This term is only used as such in Tivoli Workload Scheduler for z/OS, but the concept applies also to the distributed product. See also:
- offset-based run cycle
- run cycle

run cycle. Specifies the days that a job stream is scheduled to run. See also:
- calendar
- exclusive run cycle
- iCalendar
- inclusive run cycle
- rule-based run cycle
- simple run cycle
- weekly run cycle

S

schedule. See [job stream].

scheduled time. The time when a job or job stream is scheduled to run. See also:
- actual start time
- earliest start time
- latest start time
- planned start time

security file. The file where access rights of users to objects in the database and the plan are defined. It is created by makesec. See also [makesec].

simple run cycle. A specific set of user-defined days a job stream is run. A simple run cycle is defined for a specific job stream and cannot be used by other job streams. See also [run cycle].

standard agent. An installed agent component in a distributed Tivoli Workload Scheduler network that runs jobs, but requires a domain manager to resolve local dependencies and launch the jobs.

status. The current job or job stream status within the Job Scheduling Console. The Job Scheduling Console status is common to Tivoli Workload Scheduler and Tivoli Workload Scheduler for z/OS. See also [internal status].

successor. A job that cannot start until all of the predecessor jobs or job streams on which it is dependent are completed successfully. See also: [predecessor].

Symphony file. A file containing the scheduling information needed by the production control process (batchman) to run the plan. The file is built and loaded when the production plan is created or extended on the
master domain manager. During the production phase, it is continually updated to indicate the current status of production processing: work completed, work in progress, and work to be done. To manage production processing, the contents of the Symphony file (plan) can be displayed and altered using conman or the Job Scheduling Console. See also:

- `batchman`
- `conman`
- `job scheduling console`
- `plan`

**table view.** A graphical view in the Job Scheduling Console used to display database and plan object data in tabular format. See also:

- `database`
- `job scheduling console`
- `plan`
- `views`

**timeline view.** A graphical view in the Job Scheduling Console used to modify and maintain job stream instance time restrictions. See also:

- `job scheduling console`
- `time restriction`
- `views`

**time restriction.** Determines the times before which, after which, or both, that a job or job stream cannot be run. Specifying both defines a time frame within which a job or job stream runs. Jobs can also have a repetition rate. For example, Tivoli Workload Scheduler can launch the same job every 30 minutes between the hours of 8:30 a.m. and 1:30 p.m.

**trial plan.** A projection of the current production plan for a different period, using the same start date. It is used to determine the effect of different plan decisions. See also `plan`.

**user options.** Configuration options defined for each user on a workstation, in a useropts file for each user on a workstation. The settings in this file apply only to that user on that workstation. See also:

- `local options`
- `global options`

**utility commands.** A set of utilities invoked from the operating system’s command line for managing Tivoli Workload Scheduler.

**views.** Elements of the graphical user interface of the Job Scheduling Console used for viewing and modifying scheduling objects. See also:

- `explorer view`
- `graph view`
- `impact view`
- `table view`
- `timeline view`

**weekly run cycle.** A run cycle that specifies the days of the week that a job stream is run. For example, a job stream can be specified to run every Monday, Wednesday, and Friday using a weekly run cycle. A weekly run cycle is defined for a specific job stream and cannot be used by multiple job streams. See also `run cycle`.

**workstation.** A definition of an individual computer or computer partition on which jobs and job streams are run. Types of workstation vary depending on the type of engine. See also:

- `distributed workstation`
- `z/OS workstation`

**workstation class.** A workstation class is a group of workstations with similar job-scheduling characteristics. Any number of workstations can be placed in a class. Job streams and jobs can be assigned to run on a workstation class. This makes replication of a job or job stream across many workstations easy. See also `workstation`.

**writer.** A process started by netman. The writer process passes incoming messages to the local mailman process. The writer processes (there may be more than one on a domain manager workstation) are started by link requests and are stopped by unlink requests or when the communicating mailman process ends. See also `processes`.

**x-agent.** See `extended agent`.

**z/OS network.** A connected group of workstations that use the Tivoli Workload Scheduler z/OS engine to perform workload scheduling. See also:

- `engine`
- `workstation`
**z/OS workstation**

*z/OS workstation.* A representation of system configuration elements in the Tivoli Workload Scheduler for z/OS network. For the z/OS engine, workstations can be:

- Computer
- General
- Printer

See also [workstation](#).
Index

Numerics
5601-453, program number xx
5601-454, program number xx
5765-086, program number xx

A
accessibility xx
appserver auto restart option 152
appserver check interval option 152
appserver count reset interval option 152
appserver max restarts option 152
appserver min restart time option 152
appserver service name 152
auditing
database level 145
authorization roles
installation wizard 16, 17
Software Distribution 17
twinsist 17
autostart monman option 152

B
backup domain manager
switching 9
behind firewall
extended agent 177
behindfirewall option 177
bm check deadline option 152
bm check file option 152
bm check status option 152
bm check until option 152
bm look option 153
bm read option 153
bm status option 153
bm verbose option 153
BmEvents.conf file
CHSCHED parameter 202
configuration 201
EVENT parameter 202
FILE parameter 203
FILE_NO_UTF8 parameter 203
LOGGING parameter 201
OPTIONS parameter 201
SYMEVNTS parameter 201
BookManager xx
books
see publications xiii

C
can be event processor option 153
can only SSL auth mode option 156
CDs 21
centralized security option 144
certificates 166
certification authority 167

CLI
conman 9
switchmgr 9
wimpspo 66
winstsp 66
commands
console 162
link 10
optman 143
switchmgr 9, 154
unlink 10
wdrmvsp 135
wimpspo 66
winstsp 66, 67
console command 162
console messages and prompts 161
conventions, typeface xxi
cpu SSL auth mode option 156
customer support
See Software Support

d
database audit level 145
directory names, notation xxii
domains
topology
multiple 7
single 6

e
education 237
enable list security check 146
encryption ciphers
shortcuts
EXP 157
EXPORT40 157
EXPORT56 157
HIGH 157
LOW 157
MEDIUM 157
NULL 157
SSLv3 157
TLSv 157
database level 145
directory names, notation xxii
domains
topology
multiple 7
single 6
events (continued)
numbers 213
examples
language installation 67
extended agent
behind firewall 177
fault tolerance
backup domain manager 9
switching 9
fault-tolerant switch mechanism 9
files
/etc/password 129
localopts 150
pobox 9
Sfinal 140
swdis.ini 66
Symphony 8
final job stream
adding 140
firewall support 177
fix packs
obtaining 239
fixes
See fix packs

global option descriptions
baseRecPrompt 143
carryforward 144
carryStates 143
companyName 144
deploymentFrequency 144
tenCentSec 144
tenCfinterNetworkDeps 145
tenCfResourceQuantity 145
tenDbAudit 145
tenEmptySchedsAreSucc 145
tenEventDrivenWorkloadAutomation 145
tenEventProcessorHttpsProtocol 145
tenLegacyId 145
tenLegacyStartOfDayEvaluation 146
tenListSecChk 146
tenLogonBatch 146
tenPlanAudit 146
ntenPreventStart 147
tenRetainNameOnRerunFrom 147
tenStrEncrypt 147
tenSwfaultTol 147
tenTimeZone 147
tenEventProcessorElfPort 147
tenRecPrompt 147
ignoreCals 147
logCleanupFrequency 148
logHistory 148
logmanMinMaxPolicy 148
logmanSmoothPolicy 148

© Copyright IBM Corp. 1991, 2007
IBM Redbooks 237
IBM support assistant 237
information centers
at IBM support Web site, searching for problem resolution 238
local, searching for problem resolution 238
installation
adding new features 58
CDs 21
log files 23
silent 60
software package blocks 63
upgrading 78, 96
Using the installation wizard 42
installation CDS
Tivoli Workload Scheduler Engine for AIX 21
Tivoli Workload Scheduler Engine for HP-UX on PA-RISC 22
Tivoli Workload Scheduler for HP-UX on Itanium 22
Tivoli Workload Scheduler for Linux on POWER 22
Tivoli Workload Scheduler for Linux on x86 22
Tivoli Workload Scheduler for Linux x86-64 22
Tivoli Workload Scheduler for Linux z9 and zSeries 22
Tivoli Workload Scheduler for Solaris SPARC 22
Tivoli Workload Scheduler for Solaris x64 22
Tivoli Workload Scheduler for Windows 22
Tivoli Workload Scheduler Integrations and Engine for i5/OS 22
Tivoli Workload Scheduler Windows x64 22
installation elements
company name 20
installation elements (continued)
installation path 20
master domain manager name 20
TCP/IP port number 20
this workstation name 20
TWS user name 20
TWS user password 21
validation criteria 20
Installation method
twinst 61
installation methods
ISMP 41
Launchpad 41
Software Distribution 41
twinst 41
installation wizard
authorization roles 16, 17
integration
Tivoli Storage Manager 232
Integration with IBM Tivoli Monitoring
BmEvents.conf
EVENT parameter 227
FILE parameter 228
Bulk Discovery 231
TEPCfgConfig script 223
Internet, searching for problem resolution 238
internetwork dependencies 6
J
jm look option 154
jm nice option 154
jm no root option 154
jm read option 154
job scheduling events
assigned rule 205
automated action 205
class 205
clearing event 205
correlation activity 205
description 205
fields 213
format 213
format table 213
numbers 213
severity 205
syntax table 205
K
key
Is Monitored job 201, 226
keyjob 201, 226
keysched 201, 226
knowledge bases, searching for problem resolution 238
L
language packs
installing 58, 63, 67, 129, 135
removing 133
LDAP authentication 178
library xiii
Linux user accounts 17
list permission
enable option 146
local option descriptions
appserver auto restart 152
appserver check interval 152
appserver count reset interval 152
appserver max restarts 152
appserver min restart time 152
appserver service name 152
autostart monman 152
bm check deadline 152
bm check file 152
bm check status 152
bm check until 152
bm look 153
bm read 153
bm stats 153
bm verbos 153
can be event processor 153
candly 156
cisslicerph 153
cisslserverauth 153
cisslservercertificate 153
cissltrustedd1r 153
composer prompt 153
commcmd prompt 153
cpu 156
date format 153
defaultws 154
host 154
is remote cli 154
jm job table size 154
jm look 154
jm nice 154
jm no root 154
jm read 154
local was 154
merge logtrace 154
merge stdlists 154
mm cache mailbox 154
mm cache size 154
mm resolve master 154
mm response 155
mm retry link 155
mm sound off 155
mm start tsmserver 155
mm unlink 155
mozart directory 155
nm is validated 155
nm mortal 155
nm port 156
nm read 156
nm retry 156
nm SSL port 156
parameters directory 156
port 156
proxy 156
SSL auth mode 156
SSL auth string 157
SSL CA certificate 157
SSL certificate 157
SSL certificate chain 157
SSL encryption cipher 157
SSL key 157
SSL key pwd 157
SSL random seed 158
stdlist width 158
local option descriptions (continued)
string 156
switch sym prompt 158
sync level 158
syslog local 158
tcp timeout 158
thiscpu 158	
timeout 158
unison network directory 159
wr enable compression 159
wr read 159
wr unlink 159
local options
file example 150
file template 150
setting 150
setting sysloglocal 161
syntax 150
local was option 154
log files 23

M
manuals
see publications xiii
merge logtrace option 154
merge stdlists 154
message level 162
mm cache mailbox 154
mm cache size 154
mm resolve master 154
mm response option 155
mm retry link option 155
mm sound off option 155
mm start tomserv option 155
mm unlink option 155
mozart directory 155

N
naming conventions
workstations 12
network 3
nm ipvaldate 155
nm mortal 155
nm port option 156
nm read option 156
nm retry option 156
nm SSL port option 156
notation
environment variables xxii
path names xxii
		
typeface xxii

O
online publications
accessing xiv
operating systems 20
AIX 27
HP-UX 27
i5/OS 28
Solaris 27
SuSe Linux 28
optional features
connector 58

temporary features (continued)
language packs 58
optman
security settings 143
ordering publications xx

P
parameters
-create_link 129
-lang 129
-reset_perm 129
-restart 129
-skip_usercheck 129
parameters directory 156
path names, notation xxii
port number 65
private keys 166
problem determination
describing problems 242
determining business impact 242
submitting problems 242
problem resolution 237
process messages 161
process prompts 161
processes
mdemon 179
muser 180
product
library xiii
product groups 20
program numbers
5601-453 xx
5601-454 xx
5765-086 xx
publications xiii
accessing online xix
library xiii
online xx
ordering xx

R
Redbooks, IBM 237
removing the product 133

S
security level
enabled 169
force 169
on 169
services
stopping 125
setting
LANG 203
setting the local options 150
setup programs 21
Sfinal
adding 140
silent installation 60
softcopy books xx
Software Distribution
authorization roles 17
software support 237
Software Support
contacting 241
describing problems 242
determining business impact 242
receiving weekly updates 239
submitting problems 242
SSL attributes
configuring 169
SSL auth mode option 156
SSL auth string 157
SSL CA certificate option 157
SSL certificate chain option 157
SSL certificate option 157
SSL communication
enabled 169
force 169
on 169
SSL encryption cipher option 157
SSL key option 157
SSL key pwd option 157
SSL random seed option 158
SSL security
HTTP/HTTPS 165
keystore passwords 171
overview 165
RMI/IIOP 165
topology connection 165
SSL support
configuring 170
stdlist width option 158
stopping
services 125
string SSL auth mode option 156
structure
installation CD structures 21
support 237
support assistant 237
support Web site, searching to find
software problem resolution 238
switch fault tolerance mechanism 9
switching
backup domain manager 9
fault-tolerant 9
Symphony file 8
sync level 158
synchronization levels
high 158
low 158
medium 158
syntax
winstsp 67
syslog 161
syslog local option 158
sysloglocal options
LOG_ERR 161
LOG_INFO 161
LOG_NOTICE 161
LOG_WARNING 161

T
tcp port 65
tcp timeout option 158
TEC integration
BmEvents.conf 201
setup 204
terminology
accessing Web site  xix
thiscpu  158
time zone
overview  11
Tivoli Enterprise Portal integration
BmEvents.conf file
CHSCHED parameter  227
LOGGING parameter  226
OPTIONS parameter  226
SYMEVNTS parameter  226
Tivoli Software Glossary  xix
Tivoli software information center  xix
Tivoli Storage Manager
integration  232
twinst
authorization roles  17
usage  61
typeface conventions  xxi

U
uninstalling  133
unison network directory  159
UNIX user accounts  17
unlink workstations  125
upgrading  96
user name
creating  45, 56, 65

V
variables
$MASTER  154
language
Brazilian Portuguese  67
Chinese, Simplified  67
Chinese, Traditional  67
French  67
German  67
install_dir  67
Italian  67
Japanese  67
Korean  67
Spanish  67
tws_user  67
Software Package Block
backup  66
backup_dir  65
company  65
domain  65
fresh_install  65
from_release  66
ft_agent  65
install_dir  64
master_cpu  65
pwd  65
tcp_port  65
this_cpu  65
tws_user  65
upgrade  66
symlink
twshome/bin/at  23
twshome/bin/batch  23
twshome/bin/datecalc  24
twshome/bin/jobstid  24
variables, environment, notation  xxii

W
wdrmvsp command  135
winstsp
syntax  67
workstations
  naming conventions  12
  unlinking  125
wr enable compression option  159
wr read option  159
wr unlink option  159