Rational Developer for IBM i (RDi)  
--- Advanced topic

**Featuring:** Filters, filter pools, user actions,

---

**Agenda**

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- Running Commands and Launch Configurations
- Working in a Team Environment
Customizations – tailor it for your working style

- Many different ways to customize RDi
  - Workbench layout
    - Views, actions
  - Preferences
    - Colors, fonts
    - Editors
    - Keyboard shortcuts
    - Default behavior
    - Workbench Appearance
  - User Defined Actions
    - Objects, members, jobs, IFS files and folders
  - Compile Commands
    - Customize defaults
    - Add your own
  - RSE Connections
    - Library list
    - Object library

Default RSE Perspective

- Remote systems
- Team
- Properties
- Scratchpad
- Remote Systems Detail
- Tasks
- Object Table
- Commands Log
- Editor with editor defaults
Customizing the Workbench Layout

- Drag and drop views where you want them
- Close views by clicking on “X” in view’s tab
- Open views using Window > Show View > ...
- Use right click “Dock On >…” action to change location for “Open Perspectives” and “Fast Views” bars

Dock Open Perspectives on “top left”
Closed “Remote ScratchPad” view and moved “Properties” view
Remove toolbar icons and added i5/OS Project icons
Added Outline and Commands Log as Fast views. Docked Fast view bar on right.

Editor:
- Show timestamp
- Current line highlighted yellow
- Switched from 10pt to 8pt font
Saving your Workbench Layout

- Finished customizing the perspective
  - Save it as your own perspective
  - Re-use your layout

Reset a perspective

- Want to get back to the default perspective layout?
  - Rest perspective
  - Restores the default layout
Preferences

- Central place to make all other workbench customizations
  - Window > Preferences...
  - There are a lot!
    - Preference categories are searchable

- A few key ones
  - General > Appearance (general workbench appearance)
  - General > Capabilities (enable / disable capabilities)
  - General > Keys (Keyboard shortcuts)
  - LPEX
    - All LPEX preferences that are not language specific
  - Remote Systems
    - Generic RSE preferences
  - Remote Systems > i5/OS
    - i5/OS specific RSE preferences
  - Remote Systems > Remote Systems LPEX Editor
    - LPEX language specific editor preferences

Tip: Browse through the preference pages to see what can be customized

Workbench Keyboard Shortcuts

Preference page lets you view and set keyboard shortcuts

Available actions (“Commands”) are grouped by categories
Different categories can have same shortcut

Add or remove shortcuts for a command
LPEX Parsers provide language specific features like color tokenizing.

Use “Parser Associations” to associate a parser with your own source attribute (“MyRPG” with ILErpg parser)

Parser Styles page lets you customize colors used for tokenizing
LPEX Keyboard Shortcuts

LPEX > User Key Actions page lets you set keyboard shortcuts for LPEX actions.

Enter key combination and action name and click Set

c == Ctrl
s == Shift
a == Alt

All LPEX actions are documented in the online help. Press F1 in the Action entry field on preference page.
Remote Systems Preferences

Use workbench menu Window → Preferences or View → Preferences → RSE in RSE view

Control size of (and clear) temporary file cache (files are stored in local project during editing).

Settings specific to transferring and editing IFS files (binary, text, which editor)

Use cache to restore RSE view when restoring Faster, but view might be stale
**RSE LPEX preferences**

**General IBM i settings for LPEX editor**

**IBM i language specific settings for LPEX editor**

Dialog when opening a member and autosave backup exists

---

**Agenda**

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- Running Commands and Launch Configurations
- Working in a Team Environment
User Defined Actions

- User-Defined Actions … like PDM!
  - Right-click on Objects -> Work With
    - User Actions
      - Create, delete or change user-defined actions
      - Scope them so you only see them when appropriate
    - Named Types
      - Create named types to scope actions against
      - EG. “RPG” might be RPG + RPGLE + SQLRPGLE

- … and even beyond PDM!
  - Libraries, objects and members
  - Jobs
  - IFS folders and files

Defining User Actions
Defining User Actions

PDM-like substitution variables

Get list of available variable
Press Insert variable button
Or
Content assist CTRL + spacebar

When running command
1. Need prompting
2. Single selection
3. Refresh after run
4. Invoke once
5. Show on menu

Only need user action for context menu on library objects

Creating and running a user action

Press “Create” button to create new user action

Try it:
Right click a library
New user action appears on context menu

When running command
1. Need prompting
2. Single selection
3. Refresh after run
4. Invoke once
5. Show on menu

Only need user action for context menu on library objects

Press “Create” button to create new user action

Try it:
Right click a library
New user action appears on context menu

When running command
1. Need prompting
2. Single selection
3. Refresh after run
4. Invoke once
5. Show on menu

Only need user action for context menu on library objects

Press “Create” button to create new user action

Try it:
Right click a library
New user action appears on context menu
Running a user action

ADDLIBBLE rune and adds WFLABxx as last library to library list

Command log view displays job log of RSE IBM i job

Create a User Action for Jobs

Substitution variables for Job actions
Run a User Action for Jobs

User Actions for IFS

Select: folder or file to create new action

You can choose between:
• QShell commands
• QSYS commands!

The substitution variables change, depending on command type, here is the list for QShell commands

List for QShell cmds

Similar for Local, Unix, Windows and Linux user actions
User Actions for IFS continued

The substitution variables change, depending on command type
List for IBM i commands

Work With Compile Commands

- You can change IBM or vendor supplied compile commands, and add your own
Customizing Compile Commands

Create new or edit existing

Subset of PDM substitution variables allowed

- \texttt{R} - Name of the containing selection member
- \texttt{O} - Object or member library name
- \texttt{M} - None of selected member
- \texttt{L} - Object library, from Command execution properties
- \texttt{R} - Replace object when compiling, *YES* or *NO*. From Command, *YES*
- \texttt{N} - Object or member text, in single quotes

Note: your command must contain "*EVENTF" and "\texttt{SRCMBR(member\_name)}" for RSE to be able to retrieve compile feedback.

IBM i command prompt

Brings up larger window for editing

Browse IBM i for *CMD objects*

Commands are scoped by member type
Running the new Compile Command

You will see the Error List view if the command generates an event file.

Preferences For User Actions and Compile Cmds

These are global preferences, they can also be set per RSE connection.

Preferences for user actions and compile commands:
- &O var
- &R var
- &P var
- &E var
- &H var
- &G var
- &ISJ var

These are global preferences, they can also be set per RSE connection.
Agenda

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- Running Commands and Launch Configurations
- Working in a Team Environment

RSE Connections revisited

- Represents a remote system
  - Given a name
  - Multiple connections to one system permitted

- Each connection can be customized with its own:
  - i5/OS user profile
  - Library list and environment variables
  - Command execution parameters
    - Object library for compiles
    - JOBD for batch compiles / commands
  - Filters and filter pools

- Try thinking of a connection as a development project instead of a remote system
  - Order Entry Application
  - Inventory Management Application
Customizing a Connection

Use RSE preferences or use connection unique preferences
Toggle button allows to switch between the two

The commands are run top down

Specify:
- Initial library
- Initial command

Three ADDLIBE commands will be run when connection is started
Agenda

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- Running Commands and Launch Configurations
- Working in a Team Environment

Quick Tip

*LIB Expand To → For Quick Filtering
Drilling Down

- Typically we start using the RSE by just expanding:
  - Expand Library list to see libraries on lib list
  - Expand a library to see all objects in it
  - Expand a source file to see members in it
  - Expand Home directory to see folders in \home in IFS
  - Expand a folder to see all folders and files in it
  - And so on

- But often this produces lists that are too big
  - It’s unwieldy to scroll through thousands of things
  - You really want to keep lists small, to a few hundred at most

Introducing Filters

- Eventually you will need to see a subsetted list
  - Using criteria like generic names, types and attributes
  - All subsystems (nodes under a connection) support “filters”
    - Libraries, objects, members, jobs, IFS folders and files
    - These allow fine-grained control over what is shown in the RSE.

- To create a filter, right click on any subsystem and select New -> Filter Type
  - Or use the fastpath Work With Type … prompts under Objects
Introducing Filters

Pre-defined filters

New-filter actions

RSE filters for Objects

- Library filters
  - Specify simple, generic or special library names

- Object filters
  - Specify simple / generic object names, lib-qualified
    - Library name can be simple, generic or special
    - Object name can be simple or generic
    - Specify simple / generic object types and attributes
      - Can specify one or more type:attribute pairs (OR operation)

- Member filters
  - Specify simple / generic member names, lib / file-qualified
  - Specify simple / generic member types
    - Can specify one or more member types (OR operation)
Creating an object filter

Want to create filter to show all files and all programs in library RSELABxx.

Name of filter Files + PGMs in RSELABxx

Filters are really made up of 1 or more filter strings.

• You can only define one string at creation time to keep it simple.

• You can add additional filter strings in the change dialog. See next page

Changing Filters

Show all *PGM and *FILE objects in RSELABxx

Add filter criteria for object type *PGM

This allows you to create filters that capture exactly the objects or members you are interested in.
Changing Filters

Add second filter string for object type *PGM

This allows you to create filters that capture exactly the objects or members you are interested in.

Display list of filter: Show all *PGM and *FILE objects in libraries RSELABxx and WFLABxx

List doesn’t indicate library location. Object properties will display the information for a selected object.
Changing Filters

Add second filter string for object type *PGM
This allows you to create filters that capture exactly the objects or members you are interested in.

Filter Pools – More Control Over Filters

- **Eventually you will have too many filters**
  - This is a good thing, means you are using the RSE correctly

- **Time to turn on “Show Filter Pools”**
  - Filter pools are just groups of filters
  - Allow you to group filters by project, release, connection, task, etc...
  - Expanding subsystems will then first show filter pools
    - Expanding a filter pool shows filters

- RSE creates a “default filter pool”
  - All filters go here until you create your own filter pools

- Connections reference filter pools
  - Multiple connections can reference the same filter pool
Show Filter Pools changes the view to show filter pools first, then filters.

**Default Filter Pools**

**Workspace filter pool**
- Name starts: with workstation name
- Shared by all connections

**Connection filter pools**
- One per connection
- Name starts: CN-connection name
Create new Filter Pool

- In addition to using default filter pools, create your own filter pool
- When showing filter pools
  4 actions are removed from the context
  New sub menu
  2 new actions are added to the subsystem nodes

Adding filters to specific pools

New filter actions now part of context menu of filter nodes
In Show Filter Pools mode, there are two new actions on subsystems like iSeries Objects:
1. Work With Filter Pools
2. Select Filter Pools

Actions for filters:
- Delete
- Rename
- Copy
- Move

Referencing Filter Pools
Notice no weisst60 pool in this connection
Easily manage which pools this connection references
Select weisst60 pool
Referencing filter pools continued

Use New Pool Reference action
- Select the pool on the menu
Reference will be added to connection

Filter Pools, Filters and Filter Strings Summary

Hosts equals Connections
Agenda

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- **Running Commands and Launch Configurations**
- Working in a Team Environment

---

Running Commands in RSE

- There are four ways to run IBM i commands in RSE:
  - IBM i application in RSE job → runs in job the RSE connection uses
  - Batch → runs via SBMJOB (using preferences for JOBD, etc)
  - Interactive → runs in interactive job (you must do STRRSESVR)
  - Multi-threaded → runs BCI job in QUSRWRK

- You will see this prompt in many places:
  - When defining User Actions
  - When defining Compile Commands
  - When running commands in Commands Log view or Object Table view
Running Interactive Programs

- Interactive programs / commands require a 5250 emulator
  - RDI does not ship an emulator
  - Originally included i5/OS “STRRSESVR” command to associate emulator with RSE connection for running programs from RSE
    - This was added to make it easy to debug interactive applications
    - No longer required with Service Entry Points
      - See Help topic Debugging i5/OS applications

- Suggestion:
  - Don’t use STRRSESVR anymore
  - To run / debug your 5250 programs
    - Open and emulator and run them

Next two pages explain the differences
Runtime scenario separate jobs using

Interactive job for 5250
Call CLR1

RSE job and interactive job
Are not connected

RSE job with your profile

Service Entry Point
IBM i will invoke debugger thru RSE job when program starts

Launch Configurations

- Two main ways to run or debug a program
  1. Single click actions
     - Right click on Program and select one of the Run As or Debug As actions
     - Easy to use
     - Cannot specify parameters and other information
  2. Launch Configurations
     - Define all information for running or debugging your program
     - Saved so it can be easily re-run

What happens when using STRRSESVR?

Service Entry Point
IBM i will invoke debugger thru RSE job when program starts
Toolbar button for Debug and Run

- Debug actions and debug history actions

Run actions and run history actions

Launch Configurations

- Specify as many different configurations as you need
- Re-use them by selecting them from this list

Specify programs to be debugged

- Four tabs to specify info for debugging a program

- Allow debugger to debug programs accessing files in PROD libraries
  - This also a workbench preference, set it there to your preferred default value.
  - Change it here for individual special cases

Different types of launch configurations
Launch Configurations continued

- Specify command to start application to be debugged
- Prompt support for command
- Specify parameters to set the debugging environment
- Four tabs to specify info for debugging a program
- Specify source lookup path for debugger
- Default uses runtime connection
- List support to specify source path
Launch Configurations continued

Common input for different debugging environments

Share this configuration?

Display in favorites menu for fast launching?

Run configuration similar to debug

Less tabs, only How to start and Common

Specify how to launch a program give it a name and store the information for re-use

Fast launch from toolbar
Agenda

- Rational Developer for System i (RDi)
- Customizing the Workbench
- Filters and filter pools
- Running Commands and Launch Configurations
- **Working in a Team Environment**
- i5/OS Projects
- Working Disconnected

Agenda

- Customizing the Workbench
- User defined actions and compile commands
- Connections
- Filters and filter pools
- Running Commands and Launch Configurations
- **Working in a Team Environment**
Profiles

- The RSE is designed for team sharing
  - Connections
  - Filter pools
  - User-defined actions
  - Compile commands

- One person can setup environment and share with others

- Team sharing is enabled by profiles
  - All connections, filter pools, user actions are scoped per profile
    - Each profile is a folder within the RSE project
    - All data stored within subfolders
  - RSE project ("RemoteSystemsConnections") can be shared with any workbench Source Control Management provider
    - CVS, Subversion, Rational ClearCase

Team View

- RSE Team View lets you work with profiles
  - Create and delete profiles
  - Make profiles active
    - Information owned by profile shows in RSE
  - Make profiles inactive
    - Information not shown in RSE
  - Associate project with change management repository
  - Synchronize changes

Remember to backup the RemoteSystemsConnections project regularly! This contains all your RSE customizations (connections, filters, filter pools, UDA, compile commands)
Agenda

- Rational Developer for System i (RDi)
- Customizing the Workbench
- Filters and filter pools
- Running Commands and Launch Configurations
- Working in a Team Environment
- i5/OS Projects
- Working Disconnected

RSE and i5/OS Projects

- **Two foils here, more about this topic in the working offline presentation**
  - Remote System Explorer (RSE)
    - Designed to be familiar to PDE / SEU programmer
    - Remote edit, verify, compile, run / debug
    - Source members are still kept on IBM i
  - IBM i Projects
    - Designed to be similar to development of Web, Java, and XML in the workbench
    - Source is kept local on the PC in the workspace
    - Local edit and verify then push changes and build on remote system
    - Use RTCI or any workbench based SCM provider
  - Control your source and manage your projects with Rational Team Concert for IBM i
Why use IBM i projects

- Use for disconnected development
  - If you want to work on source while you’re disconnected from the System i
    - On the train
    - At home on the weekend (sorry)

- Use for structured development
  - Organize development into “projects”, just like you would for Web or Java projects
  - A project holds the required source and you build the project
  - Easy to develop and maintain versus having source in various locations
  - Source Change Management (SCM)
    - Rational Team Concert for IBM i
Summary

- Remote System Explorer provides lots of great tools for RPG and COBOL development on IBM i
- The workbench is extremely customizable
  - Each developer can customize to suite their needs and style
  - Team leads can setup RSE connections, filters, actions and compile commands and share with others
  - It takes time to learn it all

Additional Information

- Book: The Remote System Explorer: Modern Developer Tools For The System I
  By Don Yantzi and Nazmin Haji
- IBM i tools page: ibm.com/software/awdtools/iseries
- WDSC midrange.com mailing list:
  - Or email: WDSCI-L-request@midrange.com
  - Archives: http://archive.midrange.com/wdsci-l
Trademarks & Disclaimers

© IBM Corporation 1994-2008. All rights reserved.

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

- AS/400
- iSeries
- IBM
- i5/OS
- iSeries Logo
- IBM Logo
- IBM System i
- e (logo) business
- OS/400
- IBM

Trademarks & Disclaimers

Lotus, Freelance Graphics, and Word Pro are registered trademarks of Lotus Development Corporation and/or IBM Corporation.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

Lotus and all other trademarks or trade names are the trademarks or trade names of their respective owners in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Acceleron, VLSI, i5/OS, Power and Productivity are trademarks of IBM Corporation in the United States, other countries, or both.

AS/400 is a registered trademark of The Open Group in the United States and other countries.

IBM and the IBM logo are registered trademarks of IBM Corporation. Branch Services Electronic Transaction Server (BSET), System i and iSeries are registered trademarks of IBM Corporation.

Refer to IBM Publications Center or other sources for technical specifications, serviceability and warranty information.

Information is provided “AS IS” without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information in the presentation concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other public availability sources and does not constitute an endorsement by IBM of these products or services. Specific features or characteristics of non-IBM products may not be available to all customers. IBM has not tested these products and cannot confirm the accuracy of performance, capability or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. Contact your local IBM office or IBM authorized reseller for the full text of the specific Statement of Direction.

Contact information is provided here to communicate IBM’s current investment and development activities as a good faith effort to help with our customers’ future planning.

Some information in this presentation addresses anticipated future capabilities. Such information is not intended as a definitive statement of a commitment to specific levels of performance, function or delivery schedules with respect to any future products. Such commitments are only made in IBM product announcements. The information is presented here to communicate IBM’s current investment and development activities as a good faith effort to help with our customers’ future planning.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary, depending upon considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the rates stated here.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.
Summary

Remote System Explorer (RSE)
- Much more than just a replacement for SEU / PDM
- Modern application development tools for IBM i developers
- First step in learning new Eclipse based development tools
- Easily use other Eclipse tools as you need to write modern applications for IBM i and other platforms
- Gain productivity using:
  - Application Diagram Viewer
  - Outline view
  - Content assist
  - Many more

Lots of resources available to get you started:
- **RPG Café with RDi hub**
- midrange mailing list
- Webcasts and System i magazines
- COMMON and IBM Power Technical Conferences
- User group meetings