WebSphere Liberty Profile
technical overview

Manu T George
WebSphere Development
Please Note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

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 many factors, including 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 results similar to those stated here.
In the beginning (2011) there was WebSphere Application Server……

Java EE Compliant

Secure

Well-tooled

Scalable

Robust

Fast
… but customers asked for something more…..

- Lightweight
- Composable
- Simple
- Flexible
- Dynamic
- Extensible
- Cloud-ready

WebSphere Application Server
Liberty Profile

WAS 8.5, June 2012

…. but as production-ready as the original…..

…. with even higher management scale!
Lightweight

- **Java EE 6 Web Profile** certified app server in a 54MB download
Lightweight

- **Java EE 6 Web Profile** certified app server in a 53MB download
- Server starts in less than 3 seconds…
Lightweight

- **Java EE 6 Web Profile** certified app server in a 54MB download
- Server starts in less than 3 seconds…
- … with a memory footprint less than 60 MB…
Lightweight

- **Java EE 6 Web Profile** certified app server in a 54MB download
- Server starts in less than 3 seconds…
- … with a memory footprint less than 60 MB…
- … and developer tools as an Eclipse plugin
Composable

Tiny kernel with optional features

- WAS Extensions
- Java EE Implementation
- Runtime services
- Config model

Full profile

Liberty profile
- jsf
- jsp
- Security
- Servlet
- HTTP
- App Mgr
- Kernel
Composable

- Tiny kernel with optional features
- You control which features are loaded into each server instance

<feature>servlet-3.0</feature>

<feature>jsf-2.0</feature>
<feature>appSecurity-2.0</feature>

Liberty profile
Composable

► Tiny kernel with optional features
► You control which features are loaded into each server instance
► Each feature is self-contained: install and runtime

Liberty Repository

Your results for java (17 results)
Based on your filters: Feature

Managed Beans
Released: 24 April 2014
This feature enables support for the Managed Beans 1.0 specification. Managed Beans provide a common foundation for different Java EE components types that are managed by a container. Common services provided to Managed Beans include resource injection, lifecycle management and the use of interceptors.

JavaServer Faces
Released: 24 April 2014
This feature enables support for web applications that use the Java Server Faces (JSF) framework. This framework simplifies the construction of user interfaces.

Contexts and Dependency Injection
Released: 24 April 2014
The Contexts and Dependency Injection specification makes it easier to integrate Java EE components of different types. It provides a common mechanism to inject component such as EJBs or Managed Beans into other components such as JSPs or other EJBs.

Java Database Connectivity
Released: 24 April 2014
This feature enables the configuration of DataSources to access Databases from applications. Any database that complies with the JDBC 4.0 specification can be used; customized configuration of many specific providers is included. High performance connection pooling is also
Composable feature sets

Product editions contain nested sets of features

- **z/OS Edition**: Adds z/OS exploitation features
- **Network Deployment Edition**: Adds centralized management and clustering
- **“Base”, Express and Developers’ Editions**: Adds some Java EE 6 Full Profile features
- **Liberty Core Edition**: Java EE 6 Web Profile, Other core features
Composable: features in 8.5.5
Simple

- Download from wasdev.net with no login...

1. Download the Liberty profile runtime:
   ![Download Button]

2. Run the following command then follow the prompts to install the Liberty profile runtime:
   ```bash
   java -jar wlp-developers-runtime-8.5.5.2.jar
   ```
Simple

- Download from wasdev.net with no login...
- ...or let the developer tools install it for you

Use Eclipse to develop and deploy applications to Liberty profile.

1. Download and install Eclipse 4.3.2 for Java EE Developers (Kepler SR2).
2. Install WebSphere Developer Tools into Eclipse using one of the following methods:
   - **Online:** Drag and drop the Install button on to the toolbar in Eclipse, then follow the prompts:

     ![Install](Install)

   - **Offline:** Download the full update site to install offline:

     ![Download](Download)

3. In the Servers view of Eclipse, right-click to create a new Liberty profile server. Follow the steps in the wizard, which includes the option to download and install the Liberty profile runtime.
Simple

- Download from wasdev.net with no login…
- … or let the developer tools install it for you
- Install, configure a server and run an application in 1 minute
Simple

► Download from wasdev.net with no login…
► … or let the developer tools install it for you
► Install, configure a server and run an application in 1 minute
► Keep your server configuration in a single file
Simple

- Download from wasdev.net with no login…
- … or let the developer tools install it for you
- Install, configure a server and run an application in 1 minute
- Keep your server configuration in a single file
- Sparse configuration - defaults keep your config simple

```xml
<webContainer
disableAllFileServing="true">
</webContainer>
```
Simple

- Download from wasdev.net with no login…
- … or let the developer tools install it for you
- Install, configure a server and run an application in 1 minute
- Keep your server configuration in a single file
- Sparse configuration - defaults keep your config simple
- Hidden internals keep classloaders clean
Simple: future migration with Liberty profile

- There is **no migration needed for Liberty configuration**
  - the same server configurations can be used with different versions and service levels of the runtime
  - set WLP_USER_DIR to shared configuration and apps

- Existing features **will not change behavior**
  - new feature ‘versions’ will be added and will contain all updates and changes

- Supported versions of Java (J2SE) will move forward
  - Several of the Java EE 7 technologies require Java 7
  - Java 6 will go out of support at some point
    - WebSphere Application Migration Tool already has support to identify J2SE 6 -> 7 changes needed by an application
Flexible

Bring your own Java...

Any J2SE 1.6 or J2SE 7 compatible implementation
Flexible

- Bring your own Java...
- ... and your MacBook
Flexible

► Bring your own Java…
► … and your MacBook
► Structure configuration your own way….

```xml
<server description="server clone_1">
  <variable name="HTTP_default_var" value="9082" />
  <include location="http://commoncfg/global.xml" />
  <include location="${shared.config.dir}/datasource-1.xml" />
</server>
```
Flexible

- Bring your own Java...
- ... and your MacBook
- Structure configuration your own way....
- ... and maximise sharing through variables

```
HTTP_default_var=9081
```

```
bootstrap.properties file

<variable
name="HTTP_default_var"
value="9082" />
```

```
server.xml or included xml file

<httpEndpoint id="defaultHttpEndpoint"
httpPort="${HTTP_default_var}" />
```

```
export HTTP_default_var=9083
```

```
environment variable

<httpEndpoint id="defaultHttpEndpoint"
httpPort="${env.HTTP_default_var}" />
```
Flexible

- Bring your own Java...
- ... and your MacBook
- Structure configuration your own way....
- ... and maximise sharing through variables
- Install and update using our archives....

```
java -jar wlp-developers-runtime-8.5.5.1.jar
java -jar wlp-developers-runtime-8.5.5.2.jar
```
Flexible

- Bring your own Java...
- ... and your MacBook
- Structure configuration your own way....
- ... and maximise sharing through variables
- Install and update using our archives....
- ... and build your own for backup and deployment

server package myServer –include=minify

myapp.war
server.xml
servlet-3.0
kernel
Dynamic

- Dropin application install and update

Configured applications go here by default, but can be anywhere

Monitored directory for applications
Dynamic

- Dropin application install and update
- Configuration files also monitored for updates
  - All configuration changes are dynamic

Add `<application/>`

Add `<library/>`

Add `<feature>jdbc-4.0</feature>`
<dataSource/>

Modify `<dataSource>`
<properties/>
</dataSource>
Extensible

Download and install new features from Liberty Repository

Liberty Repository

Assets
Based on your filters: Feature

Sort by: Date ▼ Name ▼ Ratings

Portlet Serving
Released: 24 April 2014
This feature provides the Portlet serving or URL Addressability functionality for JSR 168 and JSR 286 compliant portlets. This feature allows to invoke a portlet under a defined URL from a browser.

Portlet Container
Released: 24 April 2014
This feature provides a runtime environment for JSR 168 and JSR 286 compliant portlets.

Web Service Security
Released: 24 April 2014
This feature provides support for securing JAX-WS web services using a WS-Security policy.

WebSphere MQ JMS Client
Released: 24 April 2014
This feature provides applications with access to message queues hosted on WebSphere MQ through the JMS 1.1 API.

http://wasdev.net/repo
Extensible

- Download and install new features from Liberty Repository
- Run inside in a java process/application

```java
Future<Result> startFuture = libertyServer.start();
Future<Result> stopFuture = libertyServer.stop();
```
Extensible

- Download and install new features from Liberty Repository
- Run inside in a java process/application
- Plugin your own features using the Liberty SPI

Package as feature or embedded server

Tools support for feature development

Install into your own extension location

Access to powerful product SPI

Manage as runtime feature, not application

OSGi management of native libraries

Integrate with server configuration

Start simple, evolve into rich solution

Expose API, services and http endpoints
Cloud-ready

- Low disk use
- Low memory use
- Very fast instance deployment
- Very fast server start
- Composable through features
- Archive install and update
- Zip file deployment
- Quiet runtime
- Most Java EE features available pre-integrated, tested, supported
- Single configuration file for simple automation
Cloud-ready: Virtual System Pattern for IaaS

VSP
- liberty WHV
- collective controller

1 to n hosts
- liberty WHV

HTTP sessions
- DB or cache

HTTP requests

Pure AS or SCO
- pattern management

IHS/plugin

PureApp Service on SoftLayer
- new
- beta
Cloud-ready: CloudFoundry PaaS

On-premise Cloud Foundry

app is packaged with runtime by liberty buildpack

+ cf push app.war

app is auto-wired to cloud services

IBM hosted
Codename: BlueMix
Beta

Elastic MQ
Session Cache
Data Cache
Log Analysis
Twilio
Mobile Data
Mongo DB
Monitoring and Analytics
SQL (DB2) Database
MySQL

Monitoring and Analytics

Cloud-Foundry PaaS
Like the WebSphere Full profile, Liberty profile remains...
Java EE (Web Profile plus) Compliant

JCA 1.6
JMS 1.1
JAX-WS 2.2
MDB 3.1

Java EE 6 Web Profile
Secure

► Out of the box…
  • localhost only, minimal ports opened
  • no default userids or passwords
  • remote admin always secured

► Registries
  • Simple registry in server.xml to get started
  • Federated LDAPs or SAF for production
  • also Custom User Registry

► OAuth 2.0
  • allows apps to delegate authorization to a third-party
Robust

- Tried and tested components reused in Liberty Profile
- Industrial strength Transaction Manager
- Failover of HTTP Sessions via DB or WXS
- Failover of Collective Controller via replica set
- Disaster recovery is similar to Full profile
  - Best practice is to have dual, independent collectives ideally in separate data centers
Fast

- Create a server and start an application in less than 1 minute.
- .... but get true production throughput

![WAS 8.5.5 Runtime Performance](image)
Fast

- Create a server and start an application in less than 1 minute....
- .... but get true production throughput
Scalable

- Liberty Collective architecture is even more highly scalable than Full Profile ND Cells

WAS Liberty test team has validated a collective of 10K servers on 5 controllers
Well-tooled

► WebSphere Developer Tools drop in to your existing eclipse
► Migration tools (WAMT) to help your move from other app servers
► Liberty Admin Center is the next-generation management tool with form factors for browser, tablet and phone
  Lightweight, task-oriented app store approach with customizable toolbox for “right-sized” UI per user / device
► Monitoring tools include:
  • IBM Application Performance Diagnostics 1.1
  • CA Wily Introspect APM 9.1.0
So what now?

- Use the WebSphere Liberty Application Tool to evaluate your applications for moving to the Liberty Profile
- Download WDT and the Liberty runtime and try it out

- Think about management options
  - Add Liberty servers to your existing WAS ND Cell
  - Build a Collective for a ‘pure liberty’ solution
  - Roll-your-own using JMX and scripting with Liberty Core Edition

- Tell us what you think, what you need
  - http://wasdev.net/answers
Questions?
Thank You