OSGi Specification Evolution

BJ Hargrave
OSGi Fellow
Core Platform Expert Group Chair
hargrave@us.ibm.com
OSGi Service Platform Release 2
Oct 2001
Planned Content

- Device Manager
- Http Service
- Log Service
- Preferences Service
- Configuration Admin
- Service Tracker
- User Admin
- Wire Admin
- XML Parser Service
- Measurement
- Position
- IO Connector Service
- Jini Service
- UPnP Service

- PackageAdmin
- PermissionAdmin
- Start Level
- URL handler support

Framework

Execution Environment
R3 Document Sections

- **Reference section**
  - Not normative
  - Intended to be illustrative, define common terminology and suggest how OSGi might be used

- **Specifications section**
  - Normative
  - Future updates to these specifications will be made backward compatible

- **Proposed Specifications section**
  - Normative
  - A proposed specification has resolved known design choices, is believed to be well understood, has received significant review and appears to enjoy enough interest to be of considerable value
  - However, these specifications lack real life experience by multiple companies which makes it difficult to guarantee full backward compatibility in the future.
  - Implementing the proposed specifications is recommended but implementers should consider the effects of future changes to the specification that, in worst case, will not be backward compatible
  - Deploying implementations of such standards into a disruption-sensitive environment is not recommended
Framework Enhancements

- Dynamic Import Package
  - Package may be imported after bundle resolution
  - Useful for `Class.forName` idiom

- Automatic import of `java.*` packages

- Improved native code selection algorithm

- Different default `FilePermissions` for bundle data area

- Additional `FrameworkEvent` types
• Two execution environments are specified

• OSGi/Minimum-1.0
  – Minimum execution environment that supports OSGi Framework and basic services implementations

• CDC-1.0/Foundation-1.0
  – J2ME Foundation Profile
  – Proper superset of OSGi/Minimum-1.0

• Bundles written to OSGi/Minimum-1.0 will also run on CDC-1.0/Foundation-1.0 and J2SE
EE Comparison

- J2SE
- OSGi/Minimum-1.0
- CDC-1.0/Foundation-1.0
Start Level Service

- Allows a management agent to order the startup sequence and shutdown sequence of bundles.
- Supports a virtually unlimited number of levels.
- The model for start levels is compatible with the Service Platform Release 2 specifications.
Start Level Service

Class Diagram

management bundle

gets

StartLevel

 Framework Event

 Framework Listener

 Framework Implementation

©1999-2002 OSGi, All Rights Reserved
URL Stream and Content Handlers

- Multiple bundles are allowed to register ContentHandler objects and URLStreamHandler objects.
- Existing schemes in an OSGi Service Platform cannot be overridden.
- The life cycle of bundles is supported.
- Scheme handlers and content type handlers become unavailable when the registering bundle is stopped.
- Minimal effort is required for a bundle to provide a new URL scheme or content type handler.
URL Stream and Content Handlers Class Diagram

Stream Handler implement.
creates connections of
URLConnection subclass impl.
Content Handler implement.

is called by

<<class>>
AbstractURLStream

URLStreamHandler Service

URL Stream Handler Proxy impl.

java.net.URL

URLStreamHandler

java.net.URL
StreamHandler Factory

java.net.URL Connection

Content Handler Proxy impl.

java.net.Content Handler

java.net.Content Handler Fact.

URL Stream Handler Proxy Fact. impl.

setURL

<<interface>>
URLStreamHandlerSetter

is tracked by

is found in registry by (keyed by mime)

©1999-2002 OSGi, All Rights Reserved
Wire Admin

- Provides a comprehensive mechanism to link data producing components with data consuming components in an OSGi environment.
- Contains configuration data in order to allow either party to adapt to the special needs of the wire.
- Facilitates the negotiation of the data type to be used for data transfer between producers of data and consumers of data.
  - Consumers and producers must be able to handle multiple data types for data exchanges using a preferred order.
- Separates connected parties from each other. Neither party is required to hold the service object of the other party.
- The interfaces are designed so that both parties, the Producer and the Consumer services, should be easy to implement.

©1999-2002 OSGi, All Rights Reserved
Wire Admin
Class Diagram

WireAdmin Listener impl.

<<service>>
WireAdmin Listener

WireAdmin Event

 Administers

<<service>>
WireAdmin

WireAdmin impl.

1

sends out events

listens to

0..*

0,1

maintains

0..*

Wire impl (persistent)

Producer impl.

<<service>>
Producer

0,1

update/poll

0..*

updated

0..*

Consumer impl.

<<service>>
Consumer

©1999-2002 OSGi, All Rights Reserved
XML Parser Service

- Leverages existing standards in Java based XML parsing: JAXP, SAX and DOM
- Runs unmodified JAXP code
- Easy to provide a SAX or DOM parser as well as easy to find a matching parser
- Possible to have multiple implementations of parsers available
- Parsers are likely to be extended in the future with more functionality
XML Parser Service
Class Diagram

- **SAXParser** user
  - parses with **DocumentBuilderFactory**
  - gets **Document**

- **Document** Builder user
  - gets **XMLParserActivator**
  - parses with **XMLParser**

**SAXParser**
- instant. by **SAXParserFactory**
- registered by **ParserImplementationBundle**

**XMLParserActivator**
- reads bundle META-INF

- **SAXParserImpl**
- **SubclassImpl**
- **DocumentBuilderFactory**
- **DocumentBuilderFactoryImpl**
A utility class that provides a consistent way of handling a diverse range of measurements

- Simplifies measurements calculations
- Handles unit conflict resolution
- Handles unit coercion
- Consistent handling of error calculation and unit types
- Uses SI units
Measurement Class Diagram

Measurement

| 0..n | is of unit |

Unit

State
• Utility class that provides an information object that has well defined semantics for a position
• Uses the World Geodetic System 84 as the datum
• Provides speed and track information
• Position information may have errors or may not be measurable
• Uses SI units for all measurements
• Works within the Wire Admin service
IO Connector Service

- Provides an intermediate layer that abstracts the actual protocol and devices from the using bundle
- Allows third-party bundles to extend the system with new protocols and devices
- Allows a protocol to be layered on top of lower layer protocols or devices
- Allows the selection of actual protocol/device by means of configuration data
- Compatible with existing standards
Jini Service

• Jini services are usable as OSGi services and vice versa

• Jini services should not compromise OSGi Service Platform security

• A bundle can export an OSGi service as a Jini service

• A bundle can use an OSGi service that represents a Jini service
Jini Service
Class Diagram

Importer impl.
0..* sets Jini templates for requested Jini services

Some imported Jini service
0..* gets
0..*
gets

Service Registrar

JiniDriver

Exporter impl.
some service tagged with EXPORT

Jini Driver Bundle

proxy object
0..* proxy for
1

Service Registrar proxy object
0..* proxy for
1

JiniDriver impl.

Some importable Jini Service
0..*
registered

Some importable Jini Service
0..*

Some exportable Jini service
0..*

Jini Community

©1999-2002 OSGi, All Rights Reserved
UPnP Service

- Limited to device control of UPnP
  - Aspects concerning the TCP/IP layer are not addressed
- OSGi services can be transparently made available to UPnP networks
- Possible to restrict the use of UPnP to a selection of the possible networks
- Bundles can listen to UPnP events
- Bundles can make a service available to UPnP controllers
- Bundles can control UPnP devices
Availability

• OSGi Service Platform Release 3
  – Specification on track to be publicly available in 1Q 2003
Backup
Framework

- Foundation of OSGi specification
- Allows applications to share a single Java VM
- Manages applications
  - Life cycle, Java Packages, Security, Dependencies between applications
- Service registry for collaboration
- Extensive notification mechanism
- Policy free
• Simple and small Log service for operator

• 4 Levels
  – INFO, DEBUG, WARNING, ERROR

• Automatically logs framework events in a defined way

• Other bundles can access log history
  – Management bundle
  – History size implementation dependent
Log Service
Class Diagram

- A log user
  - A log user bundle
  - Log a message
  - LogService
    - LogEntry
      - LogListener
        - LogReader
          - LogReader Service
            - A log reader using bundle
              - Retrieve log
  - A log service impl.
    - Store a message for retrieval and broadcast
      - Log Service Impl. bundle

- A log entry impl.
  - Message log

©1999-2002 OSGi, All Rights Reserved
Http Service

- Provides web access to bundles
- A powerful servlet runner
  - Supports Servlet Version 2.1
- Very simple to export static pages and files (like images)
- Automatically unregisters servlets when bundle is stopped
Device Access

- Dynamic device driver download model
- Plug & Play
  - Plugged in devices identify themselves
  - Device Manager will download appropriate bundle
- Matching process for best driver
- Extendable
  - Driver locator
Device Access

1. Insert camera
2. register
3. Select device
4. find bundle location
5. Install bundle
6. register
7. attach
8. Register camera
9. Select camera
10. Show camera on TV

Camera
IEEE 1394B Network Interface
TV bundle
TV
Network bundle
Device Manager
Driver Locator
Driver
Sony CCD654
Package Admin

- Policy interface to Framework for package administration
- Uninstalling a bundle will not withdraw exported packages
  - anymore (this was optional in Release 1)
- Inspect current state of package sharing
- Cleanup stale exported packages by refreshing the minimum set of bundles
Permission Admin

- Policy interface to Framework for permission administration
- Permissions associated with bundle location
  - Allows setting before bundle is downloaded
- Synchronous Bundle Listener added
  - A management agent can set the permissions Just In Time
- (Simple) Serializable format for permissions
Service Tracker

• Services register and unregister all the time
• Onus on the programmer to only use services that are registered
• ServiceTracker simplifies this task:
  – Maintains a list of active services
  – Events can be overridden in subclass
• Used in almost every bundle
Configuration Admin

- Configures bundles
  - At startup, or any later moment

- Maintains a repository of configurations
  - Local
  - Management system

- Configurations are key/value pairs
  - Can be typed with Meta Types

- Can be extended with plugins
Meta Typing

- Data Descriptors for generic editors
  - Configurations, Properties

- Supports
  - Basic types like String, Integer, Byte, Short ...
  - Arrays and Vectors

- Uses LDAP Objectclass, attribute model

- Can be localized for different languages

- General validation support
 Preferences Service

- Simple hierarchical model like Windows Registry
- Uses simple hierarchical names
  - /bundle/121/httpport=81
- Different trees
  - Multiple named trees per bundle
  - One system tree per bundle
- Implementation can store locally or on management system
Preferences Service

```
root
  ├── c1
  │    ├── d1
  │    │    └── d1
  │    │         ├── foo=8
  │    │         │    └── bar=9
  │    │         │         ├── lex='acme'
  │    │         └── d2
  │    │             └── properties /c1/d2
  │    └── c2
  │         └── /c2
  └── d1

properties /c1/d2
  └── /c1/d2/d1/d2

foo=8
bar=9
lex='acme'
```

©1999-2002 OSGi, All Rights Reserved
User Admin

- Repository of users
- Maintains data for authentication and other purposes
  - Private keys, passwords, bio-profile, User Preferences
- Powerful role based authorization model
  - Users, group of users, and groups of groups
- Administrative functions