JSR 289 Transparency

Mihir Kulkarni

JSR 289 Spec Co-Lead

September 2008
JSR 289 Specification

- SIP Servlet Specification v1.1
  - Enhance the existing SIP Servlet specification with new requirements determined by the industry
- Expert Group - 24 Companies and 34 Participants
- Major players – Oracle, IBM, ATT, Ubiquity, Ericsson, Telcordia, Cingular, Sun, Cisco, Apache, T-Mobile, 8x8, Orange, RedHat
- Over 90 Requirements discussed for JSR 289

- Timeline
  - Expert Group formation: Jan 2006
  - Public Review (PR): Jan 2008
  - Final Approval Ballot (FAB): July 2008
Transparency with developer community

- sipservlets@googlegroups.com
  - Forum to discuss SIP Servlets technology in general and SIP Servlet Specification v1.1 (JSR 289) in particular
  - Collect feedback from the developer community
  - Provide clarifications to the Spec and API
  - Foster open communication between the SIP servlets developer community and the Expert Group

- Archived at: http://groups.google.com/group/sipservlets
- Resolved issues published at: http://groups.google.com/group/sipservlets/web/pr---issues
- Current membership - 223 members
Transparency within Expert Group

- **www.289eg.org**
  - Public website hosting collaborative software – CodeBeamer – from Intland Software ([http://www.intland.com](http://www.intland.com))
  - Empower Expert Group members
    - File new issues
    - Track issue progress and resolution
    - Collaborate on proposals using Wikis
  - Empower Spec Leads
    - Efficiently manage reported issues
    - Capture discussions related to issues
    - Disburse information easily to EG members
    - Track pending work and better estimate milestone releases
289eg.org: Snapshot of task/issue tracking
6.1.2.2 **SipApplicationSession Invalidation**

There are two ways in which a SipApplicationSession can be invalidated:

1. **Explicit Invalidation mechanism**
2. **Implicit When Ready mechanism**

Once a SipApplicationSession object is invalidated by either the application or the container, it may no longer be used. All references to the object should be removed by the container and applications as soon as possible thus enabling invalidated SipApplicationSession objects to be garbage collected. The container MUST invoke the listener callback `sessionDestroyed()` if a listener exists for both SipSessions and SipApplicationSessions where they are destroyed.

---

**Edit**

Some early feedback from Sun

- We support specifying one technique, not two as written in the proposal. The `sendEvent()` and `disconnect()` approach is our preference. Is there any particular reason why we cannot pick one? Which one will be invoked? Will the container will look in the main sendor for the Sip application? Classifiers is sought in this text.

[edit 1 comments]
JSR 289 Main Features

- Application Composition
- Lifetime of SIP sessions
- ProxyBranch
- B2BUA support
- Convergence with Java EE applications
- Compliance with Java EE 5
- Multi-homing support
- Authentication of servlet initiated requests
- Session Targeting
- Parameterable interface
- servletInitialized callback
- Support for additional RFCs