

One Pager: GlassFish v2 or Sun Java System Application Server (SJSAS) 9.1

Table of Contents

[1. Introduction](#)

[1.1 Project/Component Working Name:](#)

[1.2 Name\(s\) and e-mail address of Document Author\(s\)/Supplier](#)

[1.3. Date of This Document](#)

[2. Project Summary](#)

[2.1 Project Description](#)

[2.2 Risks and Assumptions](#)

[3. Problem Summary](#)

[3.1 Problem Area](#)

[3.2 Justification](#)

[4. Technical Description](#)

[4.1 Details](#)

[4.2 Bugs/RFE's](#)

[4.3 Scope](#)

[4.4 Out-of-scope](#)

[4.5 Interfaces](#)

[4.6 Documentation Impact](#)

[4.7 Configuration/administration Impact](#)

[4.8 High Availability Impact](#)

[4.9 Internationalization](#)

[4.10 Packaging](#)

[4.11 Security Impact](#)

[4.12 Compatibility](#)

[4.12 Dependencies](#)

[5. References](#)

[6. Schedule](#)

1. Introduction

1.1. Project/Component Working Name

GlassFish v2 or Sun Java System Application Server (SJSAS) 9.1

1.2. Name(s) and e-mail address of Document Author(s)/Supplier

Sreeram Duvur (Sreeram.Duvur@sfbay.sun.com)

Dhiru Pandey (Dhiru.Pandey@sun.com)

1.3. Date of This Document

13 October 2006

2. Project Summary

2.1. Project Description

GlassFish v2 (Version 2)/SJSAS 9.1 builds upon Project GlassFish v1 (Version 1)/SJSAS 9.0 Platform Edition (PE), which was the first Java EE 5 compatible open source Application Server. This project will open source Application Server Enterprise Edition (EE) features like clustering, high availability, multi-machine administration capabilities etc. It will also integrate Java Business Integration technology (also known as Project Shasta, <http://openesb.dev.java.net>) and Web Services Interoperability Technology (also known as Project Tango <http://wsit.dev.java.net>) in the Application Server. Performance improvements in areas like SPECjAppServer2004 benchmark, startup and deployment are also goals for this release

2.2. Risks and Assumptions

As for any project, schedule and resourcing risks exist for this project. Timely availability of production quality WSIT (Web Services Interoperability Technology aka Project Tango) and JBI (Java Business Integration) technologies are assumed. We are assuming that continuous visibility of ready to use features, promoted builds and feedback from GlassFish community, will help us deliver on schedule. While we have good confidence to hit performance goals, intermediate performance gains will have to be monitored to ensure that we reach the challenging goal.

3. Problem Summary

3.1. Problem Area

Currently there are no Java EE 5 compatible open source implementations of an Application Server that provide clustering and high availability features. This project is intended to fill that gap in the open source.

3.2. Justification

Java EE 5 is gaining rapid developer adoption, due to its ease of development features. While the already released GlassFish v1 (or SJS AS/PE 9.0) delivers a free for deployment Java EE 5 container, many customers are asking for Project GlassFish to deliver a clustering capable enterprise class product, in open source. So this project is essentially making all the enterprise features (clustering, high availability etc.) also available in the open source.

4. Technical Description

4.1. Details

The main goal of this project is to make all the enterprise features of the Application Server available in the open source. Many of these features already exist in Sun Java System Application Server, Enterprise Edition (SJS AS/EE) 8.x. For eg. features like the Secure multi-machine administration, load balancing, session data check-pointing and failure recovery, reliable timers and transaction recovery features will be moved over to the open source. In addition to these, several new features and enhancements are also being added and will be described in separate one pagers. Another driver for this release is to further improve benchmark performance (SPECJAppserver 2004). In particular, major benchmark performance gains are expected from a true in-process JMS provider and application of Grizzly

v2/NIO based techniques in the ORB.

4.2. Bug/RFE Number(s)

Please see [IssueTracker for GlassFish](#)

4.3. In Scope

Listed below are new features and enhancements in this project:

- Implementation of an in-memory replication for Session Persistence. An SPI for integrating any State Management backing store Service implementations will also be defined.
- Group Management Service(GMS) will add heartbeat and instance life cycle events which may be used for recovery
- Integrated Web Services Interoperability Technology (WS-* Interoperability). For more details see <http://wsit.dev.java.net>
- Support for Java Business Integration (JBI) i.e. JSR-208. JBI runtime and components will be bundled and integrated in this release
- IIOP Path Load Balancing: Will use Group Membership Service to discover and adapt to cluster shape changes.
- Automated migration of timers and surrogate recovery of transactions in case of hard instance failures using GMS
- Http Load Balancer enhancements include include rule based load balancing such as weighted round robin and user defined decision.
- Implementation of Grizzly v2 the high performance transport component, that pulls this feature from the existing Grizzly HTTP Connector. Many other components will be consuming this implementation in future like ORB, JAX-WS, JMS etc. For this release ORB transport layer will be changed to use this implementation.
- Installer improvements: Single installer that will allow both developer and deployment configurations through

use of usage profiles. For this release only three usage profiles will be defined and exposed for users namely – developer, cluster and enterprise profile.

- Enhancements to connection pool for connection leak tracing and configuration properties.
- Support for JDBC 4.0 resource adaptor when the application server is using Java SE 6.0
- Integration of Sun Java System Message Queue v4.1. This will provide enhanced performance in the MDB/JMS path in the in-process mode (Application Server and the MQ broker are in the same virtual machine). This integration will also provide HA support with this release for 9.1 EE using peer-to-peer broker cluster topology and HADB
- Integration with new version of Generic JMS Resource Adapter (v1.7). This provides reliable redelivery of messages, load balancing for topic destinations in a cluster and improved monitoring.
- Support for update center that enables the installation of additional components and updates to existing components
- Enhancements to HADB administration (visible only to Java Enterprise System users)
- Support for JSR-196. This specification defines a standard service provider interface by which authentication mechanism providers may be integrated with containers.
- ECC (Elliptic Curve Cryptography) support
- Support for JKS (Java Key Store) format
- Support for “assign-groups” in Security Realm. See details at http://blogs.sun.com/swchan/entry/assign_groups
- Support for JDBCRealm. See details at http://blogs.sun.com/swchan/entry/jdbcrealm_in_glassfish
- Predefined Self Management rules
- Improved Web Services stack integration (Better JAXB and JAX-WS 2.1 performance)

- Support for multiple docroots in Web container. For details see https://glassfish.dev.java.net/issues/show_bug.cgi?id=948
- Web container adds support for %time-taken% token to list of supported token in "format" attribute of the <access-log> element in domain.xml. For details see https://glassfish.dev.java.net/issues/show_bug.cgi?id=973
- Support for dynamic attributes, property changes for access log in Web container. No server restart is required. For details see https://glassfish.dev.java.net/issues/show_bug.cgi?id=547
- Support for [Apache JServ Protocol version 1.3 \(ajp13\)](#). Details at https://glassfish.dev.java.net/issues/show_bug.cgi?id=200
- Support for virtual servers and web modules to be configured with custom Valves and Catalina listeners. For details see https://glassfish.dev.java.net/issues/show_bug.cgi?id=1109
- Grizzly SSL NIO non blocking implementation for improved performance. For details see http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=6439313
- New specification in this release
 - JSR-196
 - JSR-208 (JBI)
- Revisions of existing specifications
 - JSF 1.2 Maintenance Release
 - JSP Maintenance Release
 - JSTL Maintenance Release
 - Servlet 2.5 Maintenance Release
 - JAX-WS Maintenance Release
- Port of all 8.2 EE bug fixes

The process launching scheme is being simplified to improve start-up speed. The central Domain Administration Server (DAS) can be restarted on a different machine and will securely join active Node Agents. Http compression support is added.

Performance tuning will target improvements in SPECjAppServer2004 benchmark performance. Efficient in-Process messaging is expected to contribute significantly to improving the benchmark numbers.

4.4. Out of Scope

Role Based Administration, Cluster wide consolidated single-stream log viewing, and a few other features are unlikely to be completed.

A list of un-resourced but useful features will be published. GlassFish community members encouraged to contact module leads and volunteer.

4.5. Interfaces

// <http://www.opensolaris.org/os/community/arc/policies/interface-taxonomy/>

// describes the permitted interface taxonomy.

This is an umbrella project description. Detailed interface changes or new interface descriptions will appear in corresponding one pagers and functional specifications.

4.5.1 Exported Interfaces

| Interface | Stability | Former Stability | Comments |
|---|-----------|------------------|---|
| Java EE 5 (JSR-244) | Standard | Standard | We expect no incompatibilities. Mostly maintenance release of current Java EE 5 constituent JSRs are being considered. |
| Java Business Integration (JSR-208) | Standard | Standard | |

4.5.2 Imported interfaces

TBD: Import WSIT and JBI (WS-* interface list, stability levels)

| Interface | Stability | Exporting Project: Name, Specification or other Link. | Comments |
|-----------|-----------|---|----------|
| | | | |

4.6. Doc Impact

Relevant document set changes will be described in individual one pagers.

4.7. Admin/Config Impact

Only compatible extensions and enhancements are expected.

4.8. HA Impact

- Mean Time to Repair (MTTR) of DAS is now much improved
- Solaris SMF (Service Management Framework) support means better lights-out management
- Automated surrogate recovery of transactions and migration of timers will further improve service resiliency

4.9. I18N/L10N Impact

No impact.

4.10. Packaging & Delivery

Existing packaging structure is retained for Java Enterprise System. Project GlassFish will deliver installable archives in jar format.

4.11. Security Impact

No new security implications.

4.12. Compatibility Impact

No incompatibilities to declare in this document.

4.13. Dependencies

- Project Tango or WSIT
- JBI

5. Reference Documents

None

6. Schedule

[Current Schedule](#)