

Title: Java Servlet 4.0 Specification

Summary/Description: This JSR aims to expose the latest advances in HTTP to users of JavaEE as well as providing a vehicle for continual improvement in this foundational Java EE API.

Duration: 2 weeks

Section 1: Identification

Specification Lead: Shing Wai Chan

E-Mail Address: shing.wai.chan@oracle.com

Telephone Number: +1 408 276 5598

Specification Lead: Edward Burns

E-Mail Address: Edward.burns@oracle.com

Telephone Number: +1 407 458 0017

Initial Group Membership: TBD

Supporting this JSR:

Oracle

Greg Wilkins, Intalio Webtide

Stuart Douglas, RedHat

Mark Thomas, Pivotal

Section 2: Request

2.1 Please describe the proposed Specification:

The primary goal of this JSR is to expose support for the upcoming IETF standard HTTP/2 to users of the Servlet API. A secondary goal is to refresh the Servlet API with to achieve compliance with new features in HTTP 1.1 as well as responding to community input.

HTTP/2 is still in active development in an IETF working group (WG), however, it is expected to be completed well in advance of the completion date of Java EE 8. While it is generally a bad idea to track a moving target such as an actively developed IETF specification, several factors mitigate the risk in this case.

- HTTP/2 is heavily based on the SPDY prototype from Google. This prototype appears to have the backing of the WG in addition to the full support of Google.

- Browser support for SPDY is widespread with Chrome, Firefox, Safari, and Internet Explorer already present.
- The WG is closing in on Last Call as of late June 2014.
- The Java EE 8 JSRs are slated for proposed final draft in Q4CY2015. If HTTP/2 isn't done by then, something drastic will have gone wrong.

As HTTP/2 is the main driver for Servlet 4.0, it is safe to state some of its goals as our goals:

- Substantially and measurably improve end-user-perceived latency in most cases over HTTP 1.1 using TCP.
- Do not require multiple connections to a server to enable parallelism, thus improving the use of TCP especially regarding congestion control.

The following features from HTTP/2 are expected to be exposed by the Servlet 4.0 API:

- Request/Response multiplexing

This feature enables the above stated goals by introducing ideas from packet switched networking into the HTTP protocol. By breaking the HTTP communication into streams of messages composed of frames, communication that would otherwise require separate connections is possible within a single connection.

- Stream Prioritization

The preceding item enables a priority to be attached to each message, thus enabling applications to indicate the priority with which they should handle the messages.

- Server Push

The above mentioned frame-based communication enables the server to push responses to the client.

- Upgrade from HTTP 1.1

We will build on the upgrade mechanism in Servlet 3.1 that was added to enable WebSocket to also enable upgrade to HTTP/2.

In addition to exposing HTTP/2 to Java EE 8, we will use this JSR as a vehicle for continuous improvement of the API in response to ongoing user needs such as security enhancements and community issues from Jira.

2.2 What is the target Java platform? (i.e., desktop, server, personal, embedded, card, etc.)

This specification is targeted for Java EE 8 or higher platforms.

2.3 The Executive Committees would like to ensure JSR submitters think about how their proposed technology relates to all of the Java platform editions. Please provide details here for which platform editions are being targeted by this JSR, and how this JSR has considered the relationship with the other platform editions.

This specification targets the Java EE 8 Platform. It will be based on the corresponding release of the Java SE 8 platform.

2.4 What need of the Java community will be addressed by the proposed specification?

See 2.1 above.

2.5 Why isn't this need met by existing specifications?

See 2.1 above.

2.6 Please give a short description of the underlying technology or technologies:

A detailed description of Java Servlet functionality can be found in the Java Servlet 3.1 Specification, see <http://jcp.org/en/jsr/detail?id=340>.

2.7 Is there a proposed package name for the API Specification? (i.e., javapi.something, org.something, etc.)

javax.servlet

2.8 Does the proposed specification have any dependencies on specific operating systems, CPUs, or I/O devices that you know of?

No.

2.9 Are there any security issues that cannot be addressed by the current security model?

No.

2.10 Are there any internationalization or localization issues?

Java EE uses the I18N support in Java SE and defined in constituent technologies.

2.11 Are there any existing specifications that might be rendered obsolete, deprecated, or in need of revision as a result of this work?

The previous revision of Servlet is obsoleted by this one.

It is expected that the JAX-RS and WebSocket specifications will want to take advantage of HTTP/2 new features. These specifications will need to place requirements on Servlet 4.0 to enable using these features.

Servlet 4.0 will provide a means to detect if the user agent supports HTTP/2 so that developer can decide to use new HTTP/2 features exposed in the Servlet API.

2.12 Please describe the anticipated schedule for the development of this specification.

Q3 2014 Expert Group formed

Q1 2015 Early Draft

Q3 2015 Public Review

Q4 2015 Proposed Final Draft

Q3 2016 Final Release

2.13 Please describe the anticipated working model for the Expert Group working on developing this specification.

The primary means of communication will be email, with conference calls and face-to-face meetings scheduled as needed. We will solicit feedback from the community and leverage the open source development model.

2.14 Provide detailed answers to the transparency checklist, making sure to include URLs as appropriate:

The <http://java.net/projects/servlet-spec> project site will be used to track all issues and disseminate information on the progress of the JSR.

- Is the schedule for the JSR publicly available, current, and updated regularly?

The schedule will be available on the project site, <http://java.net/projects/servlet-spec>, and via the Community tab of the JSR. The official @servlet_spec twitter will be used for infrequent status updates.

- Can the public read and/or write to a wiki for the JSR?

The <http://java.net/projects/servlet-spec> project users list (users@servlet-spec.java.net) is used for this purpose.

- Is there a publicly accessible discussion board for the JSR that you read and respond to regularly?

The <http://java.net/projects/servlet-spec> project users list is used for this purpose.

- Have you spoken at conferences and events about the JSR recently?

Yes, at JavaOne.

- Are you using open-source processes for the development of the RI and/or the TCK?

The RI is being done under the open source GlassFish project at <http://glassfish.java.net>. The TCK is not open source.

- What are the Terms of Use required to use the collaboration tools you have prepared to use with the Expert Group, so that prospective EG members can judge whether they are compatible with the JSPA?

The terms of use are those of java.net: <http://www.java.net/javanet-web-site-terms-use>

- What is the location of your publicly-accessible Issue list? In order to enable EC members to judge whether Issues have been adequately addressed, the list must make a clear distinction between Issues that are still open, Issues that have been deferred, and those that are closed, and must indicate the reason for any change of state.

The publicly-accessible Issue list is available at http://java.net/jira/browse/SERVLET_SPEC

- What is the mechanism for the public to provide feedback on your JSR?

The public can provide feedback on the JSR by means of the users list.

- Where is the publicly-accessible document archive for your Expert Group?

The publicly-accessible document archive is available at <http://java.net/projects/servlet-spec/downloads>

- Does the Community tab for my JSR have links to and information about all public communication mechanisms and sites for the development of my JSR?

Yes, it will.

- Do you have a Twitter account or other social networking feed which people can follow for updates on your JSR?

Yes, @servlet_spec

- Which specific areas of feedback should interested community members (such as the Adopt-a-JSR program) provide to improve the JSR (please also post this to your Community tab)?

The formation of Adopt-a-JSR groups to provide technical feedback for Servlet 4.0 is encouraged and supported.

2.15 Please describe how the RI and TCK will be delivered, i.e. as part of a profile or platform edition, or stand-alone, or both. Include version information for the profile or platform in your answer.

The RI and TCK for Servlet 4.0 will be delivered in the same way they were delivered for Servlet 3.1.

2.16 Please state the rationale if previous versions are available stand-alone and you are now proposing in 2.13 to only deliver RI and TCK as part of a profile or platform edition (See sections 1.1.5 and 1.1.6 of the JCP 2 document).

N/A

2.17 Please provide the full text of the licenses that will apply to your Final Release Specification, Reference Implementation, and Technology Compatibility Kit, or provide links to the same.

TBD. No changes are expected from previous release.

2.18 Please describe the communications channel you have established for the public to observe Expert Group deliberations, provide feedback, and view archives of all Expert Group communications

The Expert Group will conduct business on a publicly readable alias. The public will have an alias on which to provide feedback and discuss issues related to the JSR. There will also be a publicly accessible JIRA and document archive. (See also 2.19 and 2.20 below.)

2.19 What is the URL of the Issue Tracker that the public can read, and how does the public log issues in the Issue Tracker?

https://java.net/jira/browse/SERVLET_SPEC

2.20 Please provide the location of the publicly accessible document archive you have created for the Expert Group.

<https://java.net/projects/servlet-spec/downloads>

Section 3: Contributions

3.1 Please list any existing documents, specifications, or implementations that describe the technology. Please include links to the documents if they are publicly available.

Java Servlet 3.1 Specification: <http://jcp.org/en/jsr/detail?id=340>.

3.2 Explanation of how these items might be used as a starting point for the work.

These specifications will be the starting point for this work. Specifically, all the source files from which the spec is generated will be re-used and modified.