

Title: Java API for RESTful Web Services (JAX-RS 2.1) Specification.

Summary/Description: This JSR is to develop JAX-RS 2.1, the next release of Java API for RESTful Web Services.

Duration: 2 weeks

Section 1: Identification

Specification Lead: Santiago Pericas-Geertsen

E-Mail Address: santiago.pericasgeertsen@oracle.com

Telephone Number: +1 561 214 4736

Specification Lead: Marek Potociar

E-Mail Address: marek.potociar@oracle.com

Telephone Number: +420 221 438 874

Initial Group Membership: TBD

Supporting this JSR: TBD

Section 2: Request

2.1 Please describe the proposed Specification:

Server-Sent Events (SSE) is a new technology defined as part of the HTML5 set of recommendations for a client (e.g., a browser) to automatically get updates from a server via HTTP. It is commonly employed for one-way streaming data transmissions in which a server updates a client periodically or every time an event takes place.

JAX-RS 2.0 introduced the notion of asynchronous processing for both the client and the server APIs. However, asynchronous processing alone cannot deliver on all the promises of a modern architecture without the help of non-blocking I/O. If only blocking I/O is available, asynchronous processing simply

pushes the problem from one thread to the next --this is akin to borrowing from a person to pay another, the problem is not really solved, only deferred. Thus, support for non-blocking I/O is necessary to achieve high throughput and efficiently manage resources like threads.

In summary, the following is a list of the tasks in scope for JAX-RS 2.1:

- Adding support for SSE.
- Improving integration with CDI.
- Exploring support for non-blocking I/O in providers (filters, interceptors, etc.).
- Evaluating ways in which declarative security can be supported either directly in this JSR or by leveraging other EE-platform JSRs.
- Making JAXB conditional on runtimes where it is available.
- Providing integration with JSON-B.
- Building upon the hypermedia API added in version 2.0.
- Investigating the reactive programming paradigm as a way to improve the JAX-RS asynchronous client API.
- Evaluating any requirements necessary to support the use of JAX-RS resource classes as controllers in the MVC 1.0 JSR.

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

This specification is targeted for Java SE 8 or higher and 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 JSR will be available standalone. It is also targeted for inclusion in the Java EE 8 platform. Additionally, Java EE 7 products will be allowed to implement JAX-RS 2.1.

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:

See 2.1 above.

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

JAX-RS 2.1 will use the same root package name as JAX-RS 2.0, namely javax.ws.rs.

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. This JSR will leverage existing HTTP security mechanisms, but will also explore ways to enable declarative (i.e. annotation-based) security model for securing RESTful application APIs developed using JAX-RS APIs and programming model in order to better fulfill developer requirements..

2.10 Are there any internationalization or localization issues?

This JSR will use the I18N support in Java SE.

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

No.

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

Q1 2016 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 e-mail. 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:

JAX-RS 2.1 will follow the same working model as JAX-RS 2.0 (JSR-339).

In particular, a project site (<https://jax-rs-spec.java.net/>) 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?**

Yes, the schedule will be available on the project page for the JSR at <http://jax-rs-spec.java.net>.

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

No, we'll use a public mailing list for comments instead (users@jax-rs-spec.java.net).

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

The mailing lists available at <https://java.net/projects/jax-rs-spec/lists>.

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

JAX-RS 2.0 (JSR 339) talks has been delivered at JavaOne 2013. We plan to publicly announce the JAX-RS 2.1 JSR at JavaOne 2014.

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

The Reference Implementation will be developed as part of the open source Project Jersey (<http://jersey.java.net>). The final RI will be available from the download page. 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 Java.net terms of use available at https://java.net/terms_of_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.**

All issues related to the specification will be tracked in the publicly accessible JAX-RS specification project issue tracker (https://java.net/jira/browse/JAX_RS_SPEC). The issue tracker link is also accessible from the JSR project page.

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

The user's e-mail alias users@jax-rs-spec.java.net.

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

Also available at Java.net <https://java.net/projects/jax-rs-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, see <https://jax-rs-spec.java.net/>.

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

Yes, @spericas and @marek_potociar.

- **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)?**

In general, any of the areas listed in Section 2.1. For more information, refer to the Community tab in the JSR page at jcp.org.

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 reference implementation will be made available standalone and also as part of the reference implementation for the Java EE 8 platform. The TCK will be made available standalone and as part of the Java EE CTS.

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.

No changes from licensing used in the previous JSR (JSR-339, JAX-RS 2.0) are planned. Final license text will be provided by Oracle Legal.

For reference, JSR 339 licensing was as follows:

- [Specification license](#)
- RI license
 - Commercial use

The RI will be available for commercial use under the [CDDL 1.1](#) open source license, the [GPLv2 with Classpath Exception](#) open source license, or [this RI license](#).

- Non-Commercial use

The RI will be available for non-Commercial use under the [CDDL 1.1](#) open source license or the [GPLv2 with Classpath Exception](#) open source license.

- TCK license
 - Commercial use

The TCK will be available for commercial use under [this TCK license](#).

- Non-Commercial use

As required by the Java Specification Participation Agreement (JSPA), the TCK will be licensed at no charge without support to qualified not-for-profit. The Compatibility Testing Scholarship Program will verify such qualification. Support may also be provided at no charge with approval of the scholarship board. For more information, please refer to:

<http://www.oracle.com/technetwork/java/index-137188.html>

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/JAX_RS_SPEC

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

<https://java.net/projects/jax-rs-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.

JAX-RS 2.0 specification (<https://jcp.org/en/jsr/detail?id=339>).

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

The JAX-RS 2.0 specification will be the starting point for this work.