



JAKARTA EE

Jakarta EE WebProfile

Jakarta EE Platform Team, <https://projects.eclipse.org/projects/ee4j.jakartaee-platform>

9, October 15, 2020

Table of Contents

- Copyright..... 2
- Eclipse Foundation Specification License..... 3
 - Disclaimers..... 3
- 1. Introduction..... 5
 - 1.1. Target and Rationale for the Web Profile..... 5
 - 1.2. Determining Applicable Requirements..... 6
 - 1.3. Acknowledgements for Version 6..... 7
 - 1.4. Acknowledgements for Version 7..... 7
 - 1.5. Acknowledgements for Version 8..... 8
 - 1.6. Acknowledgements for Jakarta EE 8..... 8
 - 1.7. Acknowledgements for Jakarta EE 9..... 8
- 2. Web Profile Definition..... 9
 - 2.1. Required Components..... 9
 - 2.2. Optional Components..... 9
 - 2.3. Additional Requirements..... 10
- Appendix A: Revision History..... 11
 - A.1. Changes in Final Release..... 11
 - A.1.1. Editorial Changes..... 11
 - A.2. Changes in Milestone Release Draft..... 11
 - A.2.1. Editorial Changes..... 11
- Appendix B: Related Documents..... 12

Specification: Jakarta EE WebProfile

Version: 9

Status: Final Release

Release: October 15, 2020

Copyright

Copyright (c) 2018, 2020 Eclipse Foundation.

Eclipse Foundation Specification License

By using and/or copying this document, or the Eclipse Foundation document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the Eclipse Foundation document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

- link or URL to the original Eclipse Foundation document.
- All existing copyright notices, or if one does not exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright (c) [\$date-of-document] Eclipse Foundation, Inc. <<url to this license>>"

Inclusion of the full text of this NOTICE must be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of Eclipse Foundation documents is granted pursuant to this license, except anyone may prepare and distribute derivative works and portions of this document in software that implements the specification, in supporting materials accompanying such software, and in documentation of such software, PROVIDED that all such works include the notice below. HOWEVER, the publication of derivative works of this document for use as a technical specification is expressly prohibited.

The notice is:

"Copyright (c) [\$date-of-document] Eclipse Foundation. This software or document includes material copied from or derived from [title and URI of the Eclipse Foundation specification document]."

Disclaimers

THIS DOCUMENT IS PROVIDED "AS IS," AND THE COPYRIGHT HOLDERS AND THE ECLIPSE FOUNDATION MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

THE COPYRIGHT HOLDERS AND THE ECLIPSE FOUNDATION WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of the copyright holders or the Eclipse Foundation may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

Chapter 1. Introduction

This specification defines the Jakarta™ EE Web Profile (“Web Profile”), a profile of the Jakarta™ Platform, Enterprise Edition specifically targeted at web applications.

1.1. Target and Rationale for the Web Profile

The Web Profile is targeted at developers of modern web applications.

With the term “modern” we intend to highlight the fact that the world of web applications has made much progress since the introduction of the first Servlet specification. Inevitably, the number of technologies used to create even simple web applications had grown by leaps and bounds. In fact, few web applications today are written directly to the servlet API: most applications rely on standard or third-party frameworks and libraries, often developed as open source, which in turn use the services of the servlet container.

Besides managing HTTP interactions, most web applications have significant requirements in the areas of transaction management, security and persistence. Such requirements can be readily addressed by technologies that have been part of the Jakarta EE platform for quite some time, such as the Jakarta Enterprise Beans 3.x technology and the Jakarta Persistence, but that are rarely supported by “plain” servlet containers. By incorporating many of these APIs, the Web Profile aims at raising the bar for what should be considered a basic stack for the development of web applications using the Java platform.

Targeting “modern” web applications then implies offering a reasonably complete stack, composed of standard APIs, and capable out-of-the-box of addressing the needs of a large class of web applications. Furthermore, this stack should be easy to grow, so as to address any remaining developer needs.

Against this drive towards completeness, one wishes to balance a desire to limit the footprint of web containers, both in physical and in conceptual terms. From the point of view of developers learning the Web Profile, it is more valuable to have a small, focused profile, with as little overlap between technologies as possible, rather than a more powerful but overly complex one, with redundant APIs.

In defining the Web Profile we strove to find a middle ground between these two sets of requirements.

In terms of completeness, the Web Profile offers a complete stack, with technologies addressing presentation and state management (Jakarta Server Faces, Jakarta Server Pages), core web container functionality (Jakarta Servlet), business logic (Jakarta Enterprise Beans Lite), transactions (Jakarta Transactions), persistence (Jakarta Persistence) and more.

As for simplicity, it leaves out many of the enterprise backend APIs that are part of the Jakarta EE platform. It also relies on the pluggability features in the Servlet specification to allow applications to use libraries that extend the servlet container with minimal configuration overhead.

Finally, it is worth reminding that Web Profile products are allowed to ship with more technologies

than the required ones. It is conceivable that products will offer a choice at installation time between different configurations, some richer in extensions, or even allow for complete customization beyond the required core (“à la carte” installation).

1.2. Determining Applicable Requirements



Profile definitions can be quite terse, amounting to little more than a list of required technologies and a (possibly empty) set of additional requirements, beyond those entailed by all the referenced specifications. Being the first profile of the Java™ EE 6 Platform to be defined, we expect the Web Profile specification to be used as a model for future profiles. It will also be seen as a starting point for understanding how the requirements defined in the Jakarta EE Platform specification apply to a profile that subsets the platform itself, a significant innovation in this version of the platform. (The case of a profile that is a superset of the platform is much easier to picture.) To help with this process, this section attempts to shed light on how one should go from the definition of the Web Profile to figuring out Jakarta EE 9 Web Profile, the exact set of requirements that apply to it, and consequently to any product that implements it.

As dictated by the general rules for Jakarta EE profiles in the Platform specification, products that implement the Web Profile must honor:

1. all requirements of the Jakarta EE Platform specification that apply to all profiles;
2. all requirements of this specification;
3. all requirements of the individual component specifications;
4. all requirements in the Jakarta EE Platform specification that are conditional on the presence of a specific technology or combinations of technologies.

Let’s look at some examples of requirements from each grouping.

For the first one, the Jakarta EE Platform specification mandates support for the “*java:*” naming context in all profiles. Consequently, Web Profile products must support it. For a similar reason, all Web Profile 9 APIs must support the Java™ Platform, Standard Edition 8 API, and all Web Profile 9 products must run on Java™ Platform, Standard Edition 8 runtime.

In the second category one can point out the requirement to support Jakarta EE web application modules (*.war* files) ([see Additional Requirements](#)).

The third category is hopefully self-explanatory. For example, Web Profile products must implement the Servlet API, which in turn means they need to satisfy all the requirements listed in the Jakarta Servlet specification.

The fourth category is the most complex. As a first example, since a Web Profile product is required to implement the Servlet technology, it must also follow all general requirements for Jakarta EE web containers in the Platform specification. Additionally, it must follow all security requirements in the

Platform specification that pertain to Jakarta EE web containers, all interoperability requirements for such containers, etc. Furthermore, since a Web Profile product must implement the Jakarta Transactions API, it must also satisfy all the Platform specification's transaction management-related requirements for web components, which indeed are conditional on the presence of Jakarta Servlet and Jakarta Transactions.

As a negative example for the fourth category of requirements, consider the Jakarta Messaging technology. Since it is not a required component of the Web Profile, Web Profile products are not required to include an implementation of Jakarta Messaging, nor do they have to support other Jakarta Messaging-related requirements, like the ability to inject message destination references. On the other hand, a Web Profile product that included an implementation of Jakarta Messaging would be required to honor all the Jakarta Messaging-related requirements in the Jakarta EE Platform specification.

Particular care should be taken when determining applicable requirements based on the presence of Jakarta Enterprise Beans Lite in the Web Profile. As described in the Jakarta Enterprise Beans specification, Jakarta Enterprise Beans Lite is a subset of the Jakarta Enterprise Beans API. When examining an Jakarta Enterprise Beans-related requirement in the Jakarta EE Platform spec, one must first of all determine which API classes, component types and Jakarta Enterprise Beans container services are mentioned in the requirement itself. Only if all of them fall inside the Jakarta Enterprise Beans Lite subset that requirement is considered applicable to Web Profile products.

For example, since Jakarta Enterprise Beans Lite does not include any remote functionality, the *EJB* annotation may not be used to inject a remote reference, something that should be kept in mind when evaluating the requirements in the Platform specification section “Jakarta Enterprise Beans References”.

1.3. Acknowledgements for Version 6

Version 6 of this specification was created under the Java Community Process as JSR-316. The spec leads for the JSR-316 Expert Group were Bill Shannon (Sun Microsystems, Inc.) and Roberto Chinnici (Sun Microsystems, Inc.). The expert group included the following members: Florent Benoit (Inria), Adam Bien (Individual), David Blevins (Individual), Bill Burke (Red Hat Middleware LLC), Larry Cable (BEA Systems), Bongjae Chan (Tmax Soft, Inc.), Rejeev Divakaran (Individual), Francois Exertier (Inria), Jeff Genender (Individual), Antonio Goncalves (Individual), Jason Greene (Red Hat Middleware LLC), Gang Huang (Peking University), Rod Johnson (SpringSource), Werner Keil (Individual), Michael Keith (Oracle), Wonseok Kim (Tmax Soft, Inc.), Jim Knutson (IBM), Erika S. Kohen (Individual), Peter Kristiansson (Ericsson AB), Changshin Lee (NCsoft Corporation), Felipe Leme (Individual), Ming Li (TongTech Ltd.), Vladimir Pavlov (SAP AG), Dhanji R. Prasanna (Google), Reza Rahman (Individual), Rajiv Shivane (Pramati Technologies), Hani Suleiman (Individual).

1.4. Acknowledgements for Version 7

Version 7 of this specification was created under the Java Community Process as JSR-342. The Expert Group work for this specification was conducted by means of the <http://javaee-spec.java.net> project in order to provide transparency to the Java community. The specification leads for the JSR-342 Expert

Group were Bill Shannon (Oracle) and Linda DeMichiel (Oracle). The expert group included the following members: Deepak Anupalli (Pramati Technologies), Anton Arhipov (ZeroTurnaround), Florent Benoit (OW2), Adam Bien (Individual), David Blevins (Individual), Markus Eisele (Individual), Jeff Genender (Individual), Antonio Goncalves (Individual), Jason Greene (Red Hat, Inc.), Minehiko Iida (Fujitsu), Alex Heneveld (Individual), Jevgeni Kabanov (Individual), Ingyu Kang (Tmax Soft, Inc.), Werner Keil (Individual), Jim Knutson (IBM), Ming Li (TongTech Ltd.), Pete Muir (Red Hat, Inc.), Minoru Nitta (Fujitsu), Reza Rahman (Caucho Technology, Inc), Kristoffer Sjogren (Ericsson AB), Kevin Sutter (IBM), Spike Washburn (Individual), Kyung Koo Yoon (Tmax Soft).

1.5. Acknowledgements for Version 8

Version 8 of this specification was created under the Java Community Process as JSR-366. The Expert Group work for this specification was conducted by means of the <http://javaee-spec.java.net> and <https://github.com/javaee-spec> projects in order to provide transparency to the Java community. The specification leads for the JSR-366 Expert Group were Bill Shannon (Oracle) and Linda DeMichiel (Oracle). The expert group included the following members: Florent Benoit (OW2), David Blevins (Tomitribe), Jeff Genender (Savoir Technologies), Antonio Goncalves (Individual), Jason Greene (Red Hat), Werner Keil (Individual), Moon Namkoong (TmaxSoft, Inc.) Antoine Sabot-Durand (Red Hat), Kevin Sutter (IBM), Ruslan Synytsky (Jelastic, Inc.), Markus Winkler (oparco - open architectures & consulting). Reza Rahman (Individual) participated as a contributor.

1.6. Acknowledgements for Jakarta EE 8

The Jakarta EE 8 specification was created by the Jakarta EE Platform Specification Project with guidance provided by the Jakarta EE Working Group (<https://jakarta.ee/>).

1.7. Acknowledgements for Jakarta EE 9

The Jakarta EE 9 specification was created by the Jakarta EE Platform Specification Project with guidance provided by the Jakarta EE Working Group (<https://jakarta.ee/>).

Chapter 2. Web Profile Definition

This chapter defines the contents of the Jakarta™ EE 9 Web Profile.

2.1. Required Components

The following technologies are required components of the Web Profile:

- Jakarta Servlet 5.0
- Jakarta Server Pages 3.0
- Jakarta Expression Language 4.0
- Jakarta Debugging Support for Other Languages 2.0
- Jakarta Standard Tag Library 2.0
- Jakarta Server Faces 3.0
- Jakarta RESTful Web Services 3.0
- Jakarta WebSocket 2.0
- Jakarta JSON Processing 2.0
- Jakarta JSON Binding 2.0
- Jakarta Annotations 2.0
- Jakarta Enterprise Beans 4.0 Lite
- Jakarta Transactions 2.0
- Jakarta Persistence 3.0
- Jakarta Bean Validation 3.0
- Jakarta Managed Beans 2.0
- Jakarta Interceptors 2.0
- Jakarta Contexts and Dependency Injection 3.0
- Jakarta Dependency Injection 2.0
- Jakarta Security 2.0
- Jakarta Authentication 2.0

2.2. Optional Components

There are no optional components in the Web Profile.

Web Profile products may support some of the technologies present in the full Jakarta EE Platform and not already listed in [Required Components](#), consistently with their compatibility requirements.

2.3. Additional Requirements

Web Profile products must support the deployment of Jakarta EE web application modules (*.war* files). No other modules types are required to be supported.

Appendix A: Revision History

A.1. Changes in Final Release

A.1.1. Editorial Changes

- Removed the requirement for implementations to run on Java SE 11.

A.2. Changes in Milestone Release Draft

A.2.1. Editorial Changes

- Updated all referenced Jakarta EE component versions per the Jakarta EE 9 release plan.
- Updated the requirement on Java SE 8 for the APIs and Java SE 11 for the implementations.
- Updated [“Related Documents”](#) for the updated Specifications in Jakarta EE 9.

Appendix B: Related Documents

This specification refers to the following documents. The terms used to refer to the documents in this specification are included in parentheses.

Jakarta™ EE Platform Specification Version 9. Available at: <https://jakarta.ee/specifications/platform/9>

Java™ Platform, Standard Edition, v8 API Specification (Java SE specification). Available at: <https://docs.oracle.com/javase/8/docs/>

Jakarta™ Enterprise Beans Specification, Version 4.0. Available at: <https://jakarta.ee/specifications/enterprise-beans/4.0>

Jakarta™ Server Pages Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/pages/3.0>

Jakarta™ Expression Language Specification, Version 4.0. Available at: <https://jakarta.ee/specifications/expression-language/4.0>

Jakarta™ Servlet Specification, Version 5.0. Available at: <https://jakarta.ee/specifications/servlet/5.0>

Jakarta™ Transaction Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/transactions/2.0>

Jakarta™ RESTful Web Services Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/restful-ws/3.0>

Jakarta™ Annotations Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/annotations/2.0>

Jakarta™ Debugging Support for Other Languages Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/debugging/2.0>

Jakarta™ Standard Tag Library Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/tags/2.0>

Jakarta™ Server Faces Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/faces/3.0>

Jakarta™ Persistence Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/persistence/3.0>

Jakarta™ Bean Validation Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/bean-validation/3.0>

Jakarta™ Managed Beans Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/managedbeans/2.0>

Jakarta™ Interceptors Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/interceptors/2.0>

Jakarta™ Contexts and Dependency Injection Specification, Version 3.0. Available at: <https://jakarta.ee/specifications/cdi/3.0>

Jakarta™ Dependency Injection Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/dependency-injection/2.0>

Jakarta™ WebSocket Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/websocket/2.0>

Jakarta™ JSON Processing Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/jsonp/2.0>

Jakarta™ JSON Binding Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/jsonb/2.0>

Jakarta™ Security Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/security/2.0>

Jakarta™ Authentication Specification, Version 2.0. Available at: <https://jakarta.ee/specifications/authentication/2.0>