RFP 120 - Signature Verification for Required Third Party APIs

Draft 0.2

6 Pages

Abstract

Some OSGi Alliance RFCs have a dependence on third party (non OSGi Alliance defined) APIs. Where these third party APIs are expected to be user facing, an RFC should clearly specify the set of APIs that are expected to be made available by an implementation of the corresponding specification, and the compliance tests should perform signature checking to validate that the specified APIs are present.

1 Document Information

1.1 Table of Contents

1 Document Information....2
1.1 Table of Contents....2
1.2 Terminology and Document Conventions....2
1.3 Revision History....2

2 Introduction....3

3 Application Domain....3
3.1 Terminology + Abbreviations....3

4 Problem Description....4
4.1 OSGi specification provides third-party API....4
4.2 User builds application against third-party APIs provided by OSGi specification....4
4.3 User resolves bundle with Import-Package statements for third-party API....4
4.4 Compliance test developer wants guidance on how to test third-party APIs....4

5 Use Cases....4

6 Requirements....5

7 Document Support....5
7.1 References....5
7.1 Author's Address....5
7.2 End of Document....6

1.2 Terminology and Document Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY" and "OPTIONAL" in this document are to be interpreted as described in 7.1.

```
Source code is shown in this typeface.
```

1.3 Revision History

The last named individual in this history is currently responsible for this document.

Revision	Date	Comments
Initial	*March 11th 2009*	*Adrian Colyer, SpringSource, adrian.colyer@springsource.com*
Draft 0.2	*March 24th 2009*	*Incorporated comments from Graham Charters made on eeg list*

2 Introduction

An increasing number of RFCs within the OSGi Alliance are concerned with integrating existing third-party libraries, frameworks, and APIs into the OSGi Service Platform and making these available to OSGi application developers. During the March 2009 face-to-face meeting of the Enterprise Expert Group, a discussion arose concerning the appropriate way to specify which APIs are being made available by an RFC, and how the availability of these APIs should be verified by compliance tests. This RFP addresses these questions.

3 Application Domain

This RFP is concerned with any OSGi Alliance RFC that provides or integrates third-party APIs.

3.1 Terminology + Abbreviations

- Third-party API – any API not defined in an org.osgi namespace

4 Problem Description

4.1 OSGi specification provides third-party API

An OSGi specification indicates that a third-party defined API is made available on the OSGi Service Platform. A user of the specification needs to be clear exactly what set of signatures they can rely on.

4.2 User builds application against third-party APIs provided by OSGi specification

A user creates a bundle with compile-time dependencies on a third-party API provided by an OSGi specification. The user wants to be sure that this application will compile with any compliant implementation of the specification, and the runtime libraries of the minimum specified execution environment supported by the specification, on the compilation class path.

4.3 User resolves bundle with Import-Package statements for third-party API

A user creates a bundle with run-time dependencies on a third-party API provided by an OSGi specification. The user wants to be sure that this bundle will resolve on an OSGi Service Platform at the minimum specified execution level supported by the specification, and with the specification implementation bundles installed and resolved.

4.4 Compliance test developer wants guidance on how to test third-party APIs

An implementor of a compliance test for an OSGi Alliance specification needs guidance as to how far they should go in (a) testing that the required APIs are indeed present, and (b) testing that the implementation of those APIs exhibits the desired semantics.

5 Use Cases

The following use cases should be considered as important representative cases of the problem space, rather than as an exhaustive list.

6 Requirements

- SIG001 – An RFC that makes third-party APIs available on the OSGi Service Platform MUST clearly define the set of signatures that comprise the provided third-party API
- SIG002 – A compliance test for an RFC that makes third-party APIs available on the OSGi Service Platform SHOULD perform signature-based testing to ensure that the combination of implementation plus execution environment does indeed provide the specified APIs
- SIG003 – An RFC SHOULD specify the minimum required execution environment(s) that the specification depends on. A particular implementation of the RFC may of course depend on a richer execution environment.
- SIG004 – A compliance test for an RFC that makes third-party APIs available on the OSGi Service Platform MAY perform semantic testing the API to ensure that the combination of implementation plus execution environment exhibits the behavior expected of the third-party APIs.

7 Document Support

7.1 References

[1]. Bradner, S., Key words for use in RFCs to Indicate Requirement Levels, RFC2119, March 1997.

[2]. Software Requirements & Specifications. Michael Jackson. ISBN 0-201-87712-0

7.1 Author's Address

Name	Adrian Colyer
Company	SpringSource
Address	Kenneth Dibben House, Enterprise Road, Chilworth, Southampton SO16 7NS, England.
Voice	+442380111500
e-mail	adrian.colyer@springsource.com

7.2 End of Document