PKCS #11 Cryptographic Token Interface Profiles Version 2.40

Committee Specification ~~Draft 02 /~~ ~~Public Review Draft 02~~01

~~23 April~~16 September 2014

Specification URIs

This version:

http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/cs01/pkcs11-profiles-v2.40-cs01.doc (Authoritative)
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/cs01/pkcs11-profiles-v2.40-cs01.html
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/cs01/pkcs11-profiles-v2.40-cs01.pdf

Previous version:

http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/csprd02/pkcs11-profiles-v2.40-csprd02.doc (Authoritative)
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/csprd02/pkcs11-profiles-v2.40-csprd02.html
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/csprd02/pkcs11-profiles-v2.40-csprd02.pdf

Latest version:

http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/pkcs11-profiles-v2.40.doc (Authoritative)
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/pkcs11-profiles-v2.40.html
http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/pkcs11-profiles-v2.40.pdf

Technical Committee:

OASIS PKCS 11 TC

Chairs:

Robert Griffin (robert.griffin@rsa.com), EMC Corporation
Valerie Fenwick (valerie.fenwick@oracle.com), Oracle

Editor:

Tim Hudson (tjh@cryptsoft.com), Cryptsoft Pty Ltd.

Related work:

This specification is related to:

- *PKCS #11 Cryptographic Token Interface Base Specification Version 2.40*. Edited by Susan Gleeson and Chris Zimman. Latest version. http://docs.oasis-open.org/pkcs11/pkcs11-base/v2.40/pkcs11-base-v2.40.html.
- *PKCS #11 Cryptographic Token Interface Current Mechanisms Specification Version 2.40*. Edited by Susan Gleeson and Chris Zimman. Latest version. http://docs.oasis-open.org/pkcs11/pkcs11-curr/v2.40/pkcs11-curr-v2.40.html.
- *PKCS #11 Cryptographic Token Interface Historical Mechanisms Specification Version 2.40*. Edited by Susan Gleeson and Chris Zimman. Latest version. http://docs.oasis-open.org/pkcs11/pkcs11-hist/v2.40/pkcs11-hist-v2.40.html.

- *PKCS #11 Cryptographic Token Interface Usage Guide Version 2.40*. Edited by John Leiseboer and Robert Griffin. Latest version. http://docs.oasis-open.org/pkcs11/pkcs11-ug/v2.40/pkcs11-ug-v2.40.html.

Abstract:

This document is intended for developers and architects who wish to design systems and applications that conform to the PKCS #11 Cryptographic Token Interface standard.

The PKCS #11 Cryptographic Token Interface standard documents an API for devices that may hold cryptographic information and may perform cryptographic functions.

Status:

This document was last revised or approved by the OASIS PKCS 11 TC on the above date. The level of approval is also listed above. Check the “Latest version” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=pkcs11#technical.

~~Technical Committee~~TC members should send comments on this specification to the ~~Technical Committee’s~~TC’s email list. Others should send comments to the ~~Technical Committee~~TC’s public comment list, after subscribing to it by ~~using~~following the ~~“~~instructions at the “Send A Comment” button on the ~~Technical Committee’s~~TC’s web page at https://www.oasis-open.org/committees/pkcs11/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the Technical Committee web page (https://www.oasis-open.org/committees/pkcs11/ipr.php).

Citation format:

When referencing this specification the following citation format should be used:

[PKCS11-Profiles-v2.40]

PKCS #11 Cryptographic Token Interface Profiles Version 2.40. Edited by Tim Hudson. ~~23 April~~16 September 2014. OASIS Committee Specification ~~Draft 02 / Public Review Draft 02.~~01. http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/cs01/pkcs11-profiles-v2.40-cs01.html. Latest version: http://docs.oasis-open.org/pkcs11/pkcs11-profiles/v2.40/pkcs11-profiles-v2.40.html.

Notices

Table of Contents

1 Introduction

1.1 Description of this Document

OASIS requires a conformance section in an approved committee specification ([PKCS11-Base] [TCPROC], section 2.18 Work Product Quality, paragraph 8a):

> A specification that is approved by the TC at the Public Review Draft, Committee Specification or OASIS Standard level must include a separate section, listing a set of numbered conformance clauses, to which any implementation of the specification must adhere in order to claim conformance to the specification (or any optional portion thereof).

This document intends to meet this OASIS requirement on conformance clauses for providers and consumers of cryptographic services via PKCS #11 ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance) through profiles that define the use of PKCS #11 data types, objects, functions and mechanisms within specific contexts of provider and consumer interaction. These profiles define a set of normative constraints for employing PKCS #11 within a particular environment or context of use. They may, optionally, require the use of specific PKCS #11 functionality or in other respects define the processing rules to be followed by profile actors.

For normative definition of the elements of PKCS #11 specified in these profiles, see the PKCS #11 Cryptographic Token Interface Base Specification ([PKCS11-Base]).and the PKCS #11 Cryptographic Token Interface Current Mechanisms ([PKCS11-Curr]). Illustrative guidance for the implementation of providers and consumers of PKCS #11 is provided in the PKCS #11 Cryptographic Token Interface Usage Guide ([PKCS11-UG]).

1.2 Terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in **[RFC2119]**.

1.3 Normative References

[PKCS11-Base] *PKCS #11 Cryptographic Token Interface Base Specification Version 2.40*. Edited by Susan Gleeson and Chris Zimman. 16 September 2014. OASIS Committee Specification 01. http://docs.oasis-open.org/pkcs11/pkcs11-base/v2.40/cs01/pkcs11-base-v2.40-cs01.html. Latest version: http://docs.oasis-open.org/pkcs11/pkcs11-base/v2.40/pkcs11-base-v2.40.html~~ <<DATE>>. OASIS Working Draft, .~~.

[PKCS11-Curr] *PKCS #11 Cryptographic Token Interface Current Mechanisms Specification Version 2.40*~~ <<DATE>>.~~. Edited by Susan Gleeson and Chris Zimman. 16 September 2014. OASIS ~~Working Draft,~~ Committee Specification 01. http://docs.oasis-open.org/pkcs11/pkcs11-curr/v2.40/cs01/pkcs11-curr-v2.40-cs01.html. Latest version: http://docs.oasis-open.org/pkcs11/pkcs11-curr/v2.40/pkcs11-curr-v2.40.html.

[PKCS11-Hist] *PKCS #11 Cryptographic Token Interface Historical Mechanisms Specification Version* ~~*<<VERSION>>*. <<DATE>>,~~*2.40*. Edited by Susan Gleeson and Chris Zimman. 16 September 2014. OASIS Committee Specification 01. http://docs.oasis-open.org/pkcs11/pkcs11-hist/v2.40/cs01/pkcs11-hist-v2.40-cs01.html. Latest version: http://docs.oasis-open.org/pkcs11/pkcs11-hist/v2.40/pkcs11-hist-v2.40.html~~Working Draft,~~ .

[RFC2119] Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels”, BCP 14, RFC 2119, March 1997. http://www.ietf.org/rfc/rfc2119.txt.

[TCPROC] OASIS, *Technical Committee (TC) Process, Version 31 January 2013, 31January 2013, https://www.oasis-open.org/policies-guidelines/tc-process*.

1.4 Non-Normative References

[PKCS11-UG] *PKCS #11 Cryptographic Token Interface Usage Guide ~~Specification~~ Version 2.40* ~~<<DATE>>.~~. Edited by John Leiseboer and Robert Griffin. 16 September 2014. OASIS Committee Note 01. http://docs.oasis-open.org/pkcs11/pkcs11-ug/v2.40/cn01/pkcs11-ug-v2.40-cn01.html. Latest version: http://docs.oasis-open.org/pkcs11/pkcs11-ug/v2.40/pkcs11-ug-v2.40.html~~Working Draft,~~ .

2 Profiles

2.1 PKCS #11 Profiles

This document defines a selected set of conformance clauses which form PKCS #11 Profiles. The PKCS 11 TC also welcomes proposals for new profiles. PKCS 11 TC members are encouraged to submit these proposals to the PKCS 11 TC for consideration for inclusion in a future version of this TC-approved document. However, some OASIS members MAY simply wish to inform the committee of profiles or other work related to PKCS #11.

2.2 Guidelines for Specifying Conformance Clauses

This section provides a checklist of issues that SHALL be addressed by each clause.

1. Implement functionality as mandated by **[PKCS11-Base] Section 6** (PKCS#11 Implementation Conformance)
2. Specify the list of additional data types that SHALL be supported
3. Specify the list of additional objects that SHALL be supported
4. Specify the list of additional functions that SHALL be supported
5. Specify the list of additional mechanisms that SHALL be supported

2.3 Guidelines for Validating Conformance to PKCS #11 Profiles

A PKCS #11 provider implementation SHALL claim conformance to a specific provider profile only if it instruments all required data types, objects, functions and mechanisms of that profile

- All data types specified as required in that profile
- All objects specified as required in that profile
- All functions specified as required in that profile
- All mechanisms specified as required in that profile

A PKCS #11 consumer implementation SHALL claim conformance to a specific consumer profile only if it instruments all required data types, objects, functions and mechanisms of that profile

- All data types specified as required in that profile
- All objects specified as required in that profile
- All functions specified as required in that profile
- All mechanisms specified as required in that profile

3 Conformance

3.1 Purpose of this Section

The following subsections describe currently-defined profiles related to the use of PKCS #11. The profiles define classes of PKCS #11 functionality to which an implementation can declare conformance.

3.2 Baseline Consumer Clause

A PKCS #11 consumer calls a PKCS #11 provider implementation of the PKCS #11 API in order to use the cryptographic functionality from that provider.

This profile specifies the most basic functionality that would be expected of a conformant PKCS #11 consumer – the ability to consume information via the cryptographic services offered by a provider.

3.2.1 Implementation Conformance

An implementation is a conforming Baseline Consumer Clause if it meets the conditions as outlined in the following section.

3.2.2 Conformance of a PKCS #11 Baseline Consumer

An implementation conforms to this specification as a Baseline Consumer if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the following data types:
    a. CK_VERSION ([PKCS11-Base] 3.1)
    b. CK_INFO ([PKCS11-Base] 3.1)
    c. CK_SLOT_ID ([PKCS11-Base] 3.2)
    d. CK_SLOT_INFO ([PKCS11-Base] 3.2)
    e. CK_TOKEN_INFO ([PKCS11-Base] 3.2)
    f. CK_SESSION_HANDLE ([PKCS11-Base] 3.3)
    g. CK_USER_TYPE ([PKCS11-Base] 3.3)
    h. CK_SESSION_INFO ([PKCS11-Base] 3.3)
    i. CK_OBJECT_HANDLE ([PKCS11-Base] 3.4)
    j. CK_OBJECT_CLASS ([PKCS11-Base] 3.4)
    k. CK_ATTRIBUTE_TYPE ([PKCS11-Base] 3.4)
    l. CK_ATTRIBUTE ([PKCS11-Base] 3.4)
    m. CK_RV ([PKCS11-Base] 3.6)
    n. CK_FUNCTION_LIST ([PKCS11-Base] 3.6)
    o. CK_C_INITIALIZE_ARGS ([PKCS11-Base] 3.7)
3. Supports the following objects:
    a. CKA_CLASS ([PKCS11-Base] 4.2)
    b. CKA_VALUE ([PKCS11-Base])
4. Supports the following functions:
    a. C_GetFunctionList ([PKCS11-Base] 5.4)
    b. C_Initialize ([PKCS11-Base] 5.4)
    c. C_Finalize ([PKCS11-Base] 5.4)
    d. C_GetInfo ([PKCS11-Base] 5.4)
    e. C_GetSlotList ([PKCS11-Base] 5.5)

f. C_GetSlotInfo ([PKCS11-Base] 5.5)

g. C_GetTokenInfo ([PKCS11-Base] 5.5)

h. C_OpenSession ([PKCS11-Base] 5.6)

i. C_CloseSession ([PKCS11-Base] 5.6)

5. Supports the following mechanisms:

   a. None specified

6. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
7. Optionally supports any clause within [PKCS11-Base] that is not listed above
8. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.3 Baseline Provider Clause

A PKCS #11 provider makes cryptographic functionality available to a consuming application in terms of the PKCS #11 API.

This profile specifies the most basic functionality that would be expected of a conformant PKCS #11 provider – the ability to provide information about the capabilities of the cryptographic services provided.

3.3.1 Implementation Conformance

An implementation is a conforming Baseline Provider if it meets the conditions as outlined in the following section.

3.3.2 Conformance of a PKCS #11 Baseline Provider

An implementation conforms to this specification as a Baseline Provider if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the following data types:

   a. CK_VERSION ([PKCS11-Base] 3.1)

   b. CK_INFO ([PKCS11-Base] 3.1)

   c. CK_SLOT_ID ([PKCS11-Base] 3.2)

   d. CK_SLOT_INFO ([PKCS11-Base] 3.2)

   e. CK_TOKEN_INFO ([PKCS11-Base] 3.2)

   f. CK_SESSION_HANDLE ([PKCS11-Base] 3.3)

   g. CK_USER_TYPE ([PKCS11-Base] 3.3)

   h. CK_SESSION_INFO ([PKCS11-Base] 3.3)

   i. CK_OBJECT_HANDLE ([PKCS11-Base] 3.4)

   j. CK_OBJECT_CLASS ([PKCS11-Base] 3.4)

   k. CK_ATTRIBUTE_TYPE ([PKCS11-Base] 3.4)

   l. CK_ATTRIBUTE ([PKCS11-Base] 3.4)

   m. CK_RV ([PKCS11-Base] 3.6)

   n. CK_FUNCTION_LIST ([PKCS11-Base] 3.6)

   o. CK_C_INITIALIZE_ARGS ([PKCS11-Base] 3.7)

3. Supports the following objects:

   a. CKA_CLASS ([PKCS11-Base] 4.2)

   b. CKA_TOKEN ([PKCS11-Base] 4.2)

   c. CKA_VALUE ([PKCS11-Base])

   d. CKA_ID ([PKCS11-Base])

   e. CKA_PRIVATE ([PKCS11-Base] x.y)

   f. CKA_MODIFIABLE ([PKCS11-Base)

   g. CKA_LABEL ([PKCS11-Base)

4. Supports the following functions:

    a. C_GetFunctionList ([PKCS11-Base] 5.4)
    b. C_Initialize ([PKCS11-Base] 5.4)
    c. C_Finalize ([PKCS11-Base] 5.4)
    d. C_GetInfo ([PKCS11-Base] 5.4)
    e. C_GetSlotList ([PKCS11-Base] 5.5)
    f. C_GetSlotInfo ([PKCS11-Base] 5.5)
    g. C_GetTokenInfo ([PKCS11-Base] 5.5)
    h. C_OpenSession ([PKCS11-Base] 5.6)
    i. C_CloseSession ([PKCS11-Base] 5.6)
    j. C_GetSessionInfo ([PKCS11-Base] 5.6)
    k. C_FindObjectsInit ([PKCS11-Base] 5.6)
    l. C_FindObjects ([PKCS11-Base] 5.6)
    m. C_FindObjectsFinal ([PKCS11-Base] 5.6)
    n. C_GetAttributeValue ([PKCS11-Base] 5.7)

5. Supports the following mechanisms:
    a. None specified
6. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
7. Optionally supports any clause within [PKCS11-Base] that is not listed above
8. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.4 Extended Consumer Clause

This profile builds on the PKCS#11 Baseline Consumer profile to add support for mechanism-based usage.

3.4.1 Implementation Conformance

An implementation is a conforming Extended Consumer if it meets the conditions as outlined in the following section.

3.4.2 Conformance of a PKCS #11 Extended Consumer

An implementation conforms to this specification as Extended Consumer if it meets the following conditions:

1. Supports the conditions required by the PKCS11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the conditions required by the PKCS11 Baseline Consumer clauses section 3.2
3. Supports the following additional data types:
    a. CK_MECHANISM_TYPE ([PKCS11-Base] 3.4)
    b. CK_MECHANISM ([PKCS11-Base] 3.4)
4. Supports the following additional objects:
    a. None specified
5. Supports the following additional functions:
    a. C_GetMechanismList ([PKCS11-Base] 5.5)
    b. C_GetMechanismInfo ([PKCS11-Base] 5.5)
6. Supports the following additional mechanisms:
    a. None specified
7. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
8. Optionally supports any clause within [PKCS11-Base] that is not listed above
9. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.5 Extended Provider Clause

This profile builds on the PKCS#11 Baseline Provider to add support for mechanism-based usage.

3.5.1 Implementation Conformance

An implementation is a conforming Extended Provider if it meets the conditions as outlined in the following section.

3.5.2 Conformance of a PKCS #11 Extended Provider

An implementation conforms to this specification as Extended Provider if it meets the following conditions:

1. Supports the conditions required by the PKCS #11 conformance clauses ([PKCS11-Base] Section 6 (PKCS#11 Implementation Conformance)
2. Supports the conditions required by the PKCS #11 Baseline Provider clauses section 3.3.
3. Supports the following additional data types:
    a. CK_MECHANISM_TYPE ([PKCS11-Base] 3.4)
    b. CK_MECHANISM ([PKCS11-Base] 3.4)
4. Supports the following additional objects:
    a. None specified
5. Supports the following additional functions:
    a. C_GetMechanismList ([PKCS11-Base] 5.5)
    b. C_GetMechanismInfo ([PKCS11-Base] 5.5)
    c. C_Login ([PKCS11-Base] 5.6)
    d. C_Logout ([PKCS11-Base] 5.6)
6. Supports the following additional mechanisms:
    a. None specified
7. Supports Error Handling ([PKCS11-Base] 5.1) for any supported object, function or mechanism
8. Optionally supports any clause within [PKCS11-Base] that is not listed above
9. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements

3.6 Authentication Token Clause

This profile builds on the PKCS #11 Baseline Provider and/or Baseline Consumer profiles to provide for use in the context of an authentication token.

3.6.1 Implementation Conformance

An implementation is a conforming Authentication Token if it meets the conditions as outlined in the following section.

3.6.2 Conformance of a Authentication Token

An implementation conforms to this specification as an Authentication Token if it meets the following conditions:

1. If the implementation is a consumer then it SHALL support the conditions required by the PKCS #11 Baseline Consumer Clause (Section 3.2)
2. If the implementation is a provider then it SHALL support the conditions required by the PKCS #11 Baseline Provider Clause (Section 3.3)
3. Supports the following objects:

    a. CKO_PRIVATE_KEY
    b. CKO_PUBLIC_KEY

4. Supports the following functions:
    a. C_Login
    b. C_Logout
    c. C_SignInit
    d. C_Sign and/or C_SignUpdate and C_SignFinal
5. Supports the following mechanisms:
    a. None specified
6. Optionally supports any clause within [PKCS11-Base] that is not listed above
7. Optionally supports extensions outside the scope of this standard (e.g., vendor defined extensions, conformance clauses) that do not contradict any PKCS #11 requirements.

Appendix A. Acknowledgments

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Gil Abel, Athena Smartcard Solutions, Inc.

Warren Armstrong, QuintessenceLabs

Jeff Bartell, Semper Foris Solutions LLC

Peter Bartok, Venafi, Inc.

Anthony Berglas, Cryptsoft

Joseph Brand, Semper Fortis Solutions LLC

Kelley Burgin, National Security Agency

Robert Burns, Thales e-Security

Wan-Teh Chang, Google Inc.

Hai-May Chao, Oracle

Janice Cheng, Vormetric, Inc.

Sangrae Cho, Electronics and Telecommunications Research Institute (ETRI)

Doron Cohen, SafeNet, Inc.

Fadi Cotran, Futurex

Tony Cox, Cryptsoft

Christopher Duane, EMC

Chris Dunn, SafeNet, Inc.

Valerie Fenwick, Oracle

Terry Fletcher, SafeNet, Inc.

Susan Gleeson, Oracle

Sven Gossel, Charismathics

John Green, QuintessenceLabs

Robert Griffin, EMC

Paul Grojean, Individual

Peter Gutmann, Individual

Dennis E. Hamilton, Individual

Thomas Hardjono, M.I.T.

Tim Hudson, Cryptsoft

Gershon Janssen, Individual

Seunghun Jin, Electronics and Telecommunications Research Institute (ETRI)

Wang Jingman, Feitan Technologies

Andrey Jivsov, Symantec Corp.

Mark Joseph, P6R

Stefan Kaesar, Infineon Technologies

Greg Kazmierczak, Wave Systems Corp.
Mark Knight, Thales e-Security
Darren Krahn, Google Inc.
Alex Krasnov, Infineon Technologies AG
Dina Kurktchi-Nimeh, Oracle
Mark Lambiase, SecureAuth Corporation
Lawrence Lee, GoTrust Technology Inc.
John Leiseboer, QuintessenceLabs
Sean Leon, Infineon Technologies
Geoffrey Li, Infineon Technologies
Howie Liu, Infineon Technologies
Hal Lockhart, Oracle
Robert Lockhart, Thales e-Security
Dale Moberg, Axway Software
Darren Moffat, Oracle
Valery Osheter, SafeNet, Inc.
Sean Parkinson, EMC
Rob Philpott, EMC
Mark Powers, Oracle
Ajai Puri, SafeNet, Inc.
Robert Relyea, Red Hat
Saikat Saha, Oracle
Subhash Sankuratripati, NetApp
Anthony Scarpino, Oracle
Johann Schoetz, Infineon Technologies AG
Rayees Shamsuddin, Wave Systems Corp.
Radhika Siravara, Oracle
Brian Smith, Mozilla Corporation
David Smith, Venafi, Inc.
Ryan Smith, Futurex
Jerry Smith, US Department of Defense (DoD)
Oscar So, Oracle
Graham Steel, Cryptosense
Michael Stevens, QuintessenceLabs
Michael StJohns, Individual
Jim Susoy, P6R
Sander Temme, Thales e-Security
Kiran Thota, VMware, Inc.
Walter-John Turnes, Gemini Security Solutions, Inc.
Stef Walter, Red Hat
James Wang, Vormetric
Jeff Webb, Dell

Peng Yu, Feitian Technologies

Magda Zdunkiewicz, Cryptsoft

Chris Zimman, ~~Bloomberg Finance L.P.~~Individual

Appendix B. Revision History

Revision	Date	Editor	Changes Made
wd01	20-Mar-2013	Tim Hudson	Template provided by OASIS
wd02	3-Apr-2013	Tim Hudson	Initial draft
wd03	18-Sep-2013	Tim Hudson	Updated draft matching current drafts of the specification
wd04	27-Oct-2013	Robert Griffin	Final participant list and other editorial changes for Committee Specification Draft
wd04a	27-Oct-2013	Tim Hudson	Deleted no longer valid comment and corrected unknown section reference.
csd01	30-Oct-2013	OASIS	Committeee Specification Draft
wd05	25-Feb-2014	Tim Hudson / Robert Griffin	Incorporated changes from v2.40 public review
csd02	23-Apr-2014	OASIS	Committeee Specification Draft
csd02a	Sep 3 2013	Robert Griffin	Updated revision history and participant list in preparation for Committee Specification ballot