Program Guide for Certification

Revision History

Version	Date	Notes
1.0	February 2023	• Initial Release
1.1	February 2024	• Removed Test from the title. • Added a new requirement for Certification Requester to demonstrate components are on prplWare. • Clarified that the Review Committee is for technical review. The prpl Foundation board has the final decisions on all listings.
1.2	March 2026	• Inserted sections 2.1 Preamble: The Case For prpl Certification (based on prpl's Business Requirements for Certification) and 2.2 The Use Case for Certification.

Table of Contents

Revision History 2
Table of Contents 3
1. Important notices, IPR statement, disclaimer and Copyright 5
1.1 ABOUT PRPL 5
1.2 THIS MAY NOT BE THE LATEST VERSION OF THIS PRPL DOCUMENT 5
1.3 THERE IS NO WARRANTY PROVIDED WITH THIS PRPL DOCUMENT 5
1.4 EXCLUSION OF LIABILITY 5
1.5 THIS PRPL DOCUMENT IS NOT BINDING ON PRPL NOR ITS MEMBER COMPANIES 6
1.6 INTELLECTUAL PROPERTY RIGHTS 6
1.7 COPYRIGHT PROVISIONS 6
1.7.1 INCORPORATING PRPL DOCUMENTS IN WHOLE OR PART WITHIN DOCUMENTS RELATED TO COMMERCIAL TENDERS 6
1.7.2 COPYING THIS PRPL DOCUMENT IN ITS ENTIRETY 7
2 Introduction 8
2.1 Preamble: The Case for prpl Certification (**Informative**) 8
2.2 The Use Case for Certification (**Informative**) 9
2.3 Purpose 10
2.4 Scope 10
3 References and Terminology 11
3.1 Conventions 11
3.2 Definitions 11
3.3 Certification Workflow 12
4 Certification Requester 13
4.1 Application Requirements 13
4.2 Certification Testing Requirements 13
5 Certification Tool Requirements 14
5.1 Reporting 14
5.2 Test Data 14
6 Certification Technical Review Committee 15
6.1 Application Review process 15
6.2 Committee Selection 15

7 Program Lifecycle 16
7.1 Program Guide 16
7.2 Test Plans 16
7.3 Active Certification Programs 16

1. Important notices, IPR statement, disclaimer and Copyright

This chapter contains important information about PRPL and this document (hereinafter ‘This PRPL Document’).

1.1 ABOUT PRPL

The prpl Foundation (PRPL) is a not-for-profit organization which publishes documents including, but not limited to, Requirements, Specifications, Recommendations, API application programming interfaces, and Test Plans.

1.2 THIS MAY NOT BE THE LATEST VERSION OF THIS PRPL DOCUMENT

This PRPL Document is the output of the Working Groups of the PRPL and its members as of the date of publication. Readers of This PRPL Document should be aware that it can be revised, edited or have its status changed according to the PRPL working procedures.

1.3 THERE IS NO WARRANTY PROVIDED WITH THIS PRPL DOCUMENT

The services, the content and the information in This PRPL Document are provided on an "as is" basis. PRPL, to the fullest extent permitted by law, disclaims all warranties, whether express, implied, statutory or otherwise, including but not limited to the implied warranties of merchantability, non-infringement of third-parties rights and fitness for a particular purpose. PRPL, its affiliates and licensors make no representations or warranties about the accuracy, completeness, security or timeliness of the content or information provided in the PRPL Document. No information obtained via the PRPL Document shall create any warranty not expressly stated by PRPL in these terms and conditions.

1.4 EXCLUSION OF LIABILITY

Any person holding a copyright in This PRPL Document, or any portion thereof, disclaims to the fullest extent permitted by law (a) any liability (including direct, indirect, special, or consequential damages under any legal theory) arising from or related to the use of or reliance upon This PRPL Document; and (b) any obligation to update or correct this technical report.

1.5 THIS PRPL DOCUMENT IS NOT BINDING ON PRPL NOR ITS MEMBER COMPANIES

This PRPL Document, though formally approved by the PRPL member companies, is not binding in any way upon the PRPL members.

1.6 INTELLECTUAL PROPERTY RIGHTS

Patents essential or potentially essential to the implementation of features described in This PRPL Document may have been declared in conformance to the PRPL IP Policy (available at the PRPL website: www.prplFoundation.org).

1.7 COPYRIGHT PROVISIONS

1.7.1 INCORPORATING PRPL DOCUMENTS IN WHOLE OR PART WITHIN DOCUMENTS RELATED TO COMMERCIAL TENDERS

Any or all section(s) of PRPL Documents may be incorporated into Commercial Tenders (RFP, RFT, RFQ, ITT, etc.) by PRPL and non-PRPL members under the following conditions:

(a) The PRPL Requirements numbers, where applicable, must not be changed from those within the PRPL Documents;

(b) A prominent acknowledgement of the PRPL must be provided within the Commercial document identifying any and all PRPL Documents referenced and giving the web address of the PRPL;

(c) The Commercial Tender must identify which of its section(s) include material taken from PRPL Documents and must identify each PRPL Document used, and the relevant PRPL Section Numbers; and,

(d) The Commercial Tender must refer to the copyright provisions of PRPL Documents and must state that the sections taken from PRPL Documents are subject to copyright by PRPL and/or applicable third parties.

1.7.2 COPYING THIS PRPL DOCUMENT IN ITS ENTIRETY

This PRPL Document may be electronically copied, reproduced, distributed, linked to, or made available by other means, or incorporated into or used in any information storage and retrieval system, but **only in its original, unaltered PDF format**, and with its original PRPL title and file name unaltered. It may not be modified without the advanced written permission of the PRPL.

2 Introduction

2.1 Preamble: The Case for prpl Certification (**Informative**)

Traditionally, the CPE industry used tens of different middleware “platforms” for CPEs – coming usually from ODM/OEMs. Typically, each such middleware platform got repeatedly forked in order to customize it for a particular SoC, board design and operator – which further increased the overall fragmentation. The dominant approach usually kept no compatibility between consecutive generations of CPEs – even if they were based on the same SoC vendor and provided by the same ODM/OEM. As a result, an operator having several generations of CPEs in the field and coming from a few different ODM/OEMs has tens of different middleware variants in the field. In consequence, for just a few tens of major operators, there are several hundreds of middleware variants in the field.

This huge fragmentation has a major negative impact on the operators’ capacity to innovate. For an operator having tens of different middleware variants in the field pushing a SW innovation into its CPEs is very complex and its cost is prohibitive. In fact, the cost of such adaptation, integration and testing is multiplied several times.

For an independent SW company, a business model based on offering a SW solution running in CPE is not scalable: the solution should be systematically adapted for each operator and each middleware variant. That, in-turn, fragments the solution itself, thereby making its new releases hardly manageable and costly.

Such middleware fragmentation kills not only SW-based innovation but also prohibits “data-driven” innovation. In fact, data-driven innovation requires an efficient way to collect normalized homogeneous data from all CPEs in the field. It also requires easy SW updates to tune the data generated by the CPEs and/or to tune the data-driven algorithms. None of these requirements are fulfilled with tens of different middleware variants in the field.

Due to this difficulty of innovating onto CPEs in the field, the current dominant model pushes towards faster CPE renewal with hardware innovation rather than software/data innovation. This, in-turn, increases the TCO for operators.

This huge fragmentation also makes the CPE industry less efficient. In fact, supporting tens or even hundreds of middleware variants is also difficult for SoC vendors: it increases their costs of support and bug-fixing, impacts the quality of their software deliverables and slows down the pace of updates of the Linux kernel versions. This, in turn, has a negative impact on all CPE projects: longer Time-To-Market, huge testing costs, additional back-porting costs and infrequent SW upgrades.

It is important to note that one source of detrimental fragmentation often derives from deploying various different driver versions from the SoC vendors. Such fragmentation in lower-level

software interfaces increases the costs of ODM/OEMs and system integrators working with several SoC platforms. In order to mitigate this type of fragmentation, prpl has harmonized and commonized a standard Low-Level API, based on the Linux kernel-standard interfaces, which is soon to have its own Certification Program.

Therefore, the prpl Foundation intends to address those issues. The number of middleware platform fragments should be greatly reduced. Middleware platforms will be standardized, having a common core well-delimited from well-defined customization means based on configuration, plug-ins and apps. Such a middleware design makes any upstream contributions to the common core easy because it doesn't mix improvements/extensions with customizations. This, in turn, decreases fragmentation and improves mutualization.

The middleware platform and its App Environment will also be standardized and their versioning well-managed. The marginal cost of SW innovation should become close to zero, just as has happened in all successful SW revolutions: Windows/PC, Linux/server, Android/Smartphone, etc…

The prpl Foundation has already been accomplishing all that with prplOS, LCM, HL-API, LL-API, etc… But this is not quite enough. In fact, having open-source not only doesn't prevent fragmentation, but makes it even extremely easy. Moreover, it's also quite easy to mislead operators by making false claims about following a standard open-source solution – as it used to be the case of several middleware solutions pretending to be "true OpenWrt".

Software standardization efforts already undertaken by the prpl Foundation may not be seamlessly adopted because of vested interests, bad habits and natural inertia of an industry. Therefore, the prpl Foundation decided to define and develop a strategy of formal Certification as well as some enforcement to facilitate future CPE tenders.

2.2 The Use Case for Certification (**Informative**)

Formal Certification is intended to provide assurance to Operators: For example, assurance that the middleware provided by an ODM/OEM is a full expression of authentic "prplWare" without shortcomings that could limit its intended usefulness over its lifecycle; or that the hardware with its low-level software deliverables can be relied upon to properly build and run prplWare as intended. Eventually, prpl also intends to Certify applications to provide assurance to Operators that they properly exercise prplWare and the High-Level API.

For example, Operators can issue CPE tenders citing requirements that formal Certifications of prpl's various Test Plans be obtained by their ODM/OEM vendors. If their vendors happen to produce customized derivatives from the standardized "prplWare" software, or derivatives from formally-Certified platforms, then the Operator may wish to avoid the possibility of future undesirable dependencies (e.g., so-called "vendor-proprietary lock-in"). In such cases, the

Operator is encouraged to make inquiries such as:

1) Ask for separate formal Certifications of the derivative platform;
2) Ask for the upstreaming of the derivations (e.g., bug-fixes and other modifications);
3) Ask for a full list of any/all such modifications for assessment by the Operator.

2.3 Purpose

The purpose of this document is to define the requirements for products participating in the prpl Certification Program. The document does not define specific test cases, but rather referencing how the test requirements MUST be used to achieve certification.

This document also defines other operational aspects of the prpl Certification Program, such as certification of similar products, maintenance of the program test plans, documentation, and how the certification may be used by companies with certified products.

2.4 Scope

This document applies only to operational aspects of the certification program such as:

1. What information and testing is required for products to be submitted for self certification.
2. Defining Mandatory and Optional requirements for products certification.
3. Defining the review process for certifications.
4. How companies with certified products may use the certification log and other documentation about the certification program.
5. How the test plan and program documents will be maintained and updated.
6. How companies may certify similar equipment, such as derivative products with a subset of features.

3 References and Terminology

3.1 Conventions

In these Guidelines, several words are used to signify the requirements of the specification. These words are always capitalized. More information can be found in RFC 8174 (BCP 14).

MUST	This word means that the definition is an absolute requirement.
MUST NOT	This phrase means that the definition is an absolute prohibition of the specification.
SHOULD	This word means that there could exist valid reasons in particular circumstances to ignore this item, but the full implications need to be understood and carefully weighed before choosing a different course.
SHOULD NOT	This phrase means that there could exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications need to be understood and carefully weighed.
MAY	This word means that this item is one of an allowed set of alternatives.

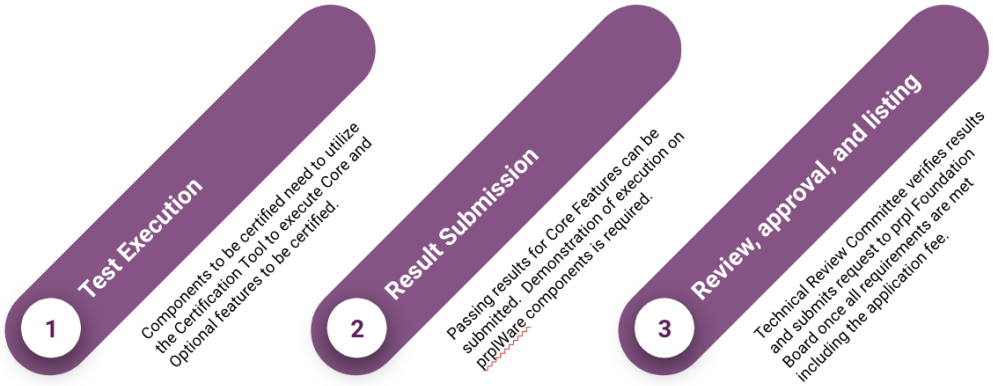
3.2 Definitions

The following terminology is used throughout these Guidelines.

Certificate	When a Certification Requester completes all the Core Features using Certification Tests with the Certification Tools.
Certification Requester	Component asking for the certification with the test results.
Certification Tool	Required tool(s) used for certification process.
Certification Tests	Test cases required for the certification.
Core Feature(s)	Test case that MUST be passed by the product applying for certification.
Optional Feature(s)	Test cases that are optional and therefore not required to achieve certification. Optional test cases might be based on features which might not be implemented on all products. Products passing optional test case(s) MAY include the optional feature(s) in their certification listing.

Certification Review Committee	A group of selected individuals to review submissions from Certification Requesters.
Major Version	Compatibility requirements may change when a prplWare component is revised. A change in compatibility requires a change in test specification.
Minor Version	Additional tests to support optional features or additional product types.
Patch Version	Test issues cause the removal or modification of tests. It shouldn't impact any previous certifications.
Test Data	Test results come from Certification Tools MUST include test tool logs and SHOULD include pcaps.

3.3 Certification Workflow



4 Certification Requester

4.1 Application Requirements

Any prpl member in good standing MAY apply for certification as long the certification request is considered an active certification program.

Test Requestors applying for certification MUST meet all the requirements defined in this section

1. MUST be a prpl member in good standing.
2. MUST complete all the additional requirements before applying for certification.
3. MUST submit all the required test results.
4. MUST include information about the product under test, including at least the product manufacturer, product model, and product software version.
5. MUST gather and submit all necessary documentation needed by the Certification Review Committee (TBD).
6. MUST demonstrate that testing was performed on prplWare components specified by individual certifications.

4.2 Certification Testing Requirements

To obtain a Certificate the following Certification Testing Requirements are:

1. Certification Requester MUST pass all the Core Feature(s) in the Certification Tests.
2. Certification Requester MUST pass all the Optional Feature(s) in the Certification Tests claimed on the application.
3. Testing MUST be completed using an approved Certification Tool (refer to Section 5).
4. Testing results MUST NOT be altered in any way.
5. Testing results MUST be reviewed and approved by Certification Review Committee.

5 Certification Tool Requirements

Certification testing MUST be completed using an approved prpl Certification Tool. Certification Tools MUST be approved by the Cert TWG. Certification Tools MUST be available independently of each certification program.

PRPL MUST maintain a list of approved certification tools. This list MUST indicate the version of the approved certification tool and the version of the certification test plan for which the tool is approved. Certification tools MUST be available for utilization by prpl members and non-members in their premises, locally or remotely, for CPE boards of any type.

Certification Tools MUST NOT limit the number of test executions in software. Certification Tools SHOULD be developed that will enable utilization of other testing or certifications, provided that it doesn't engender additional non-specifically-funded development costs.

5.1 Reporting

The report output from the Certification Tool MUST meet the following requirements:

1. MUST indicate the version of the Certification Tool used to generate the report.
2. MUST use the same test case labels and names as defined in the Certification Tests.
3. MUST be a PDF document that contains pass/fail for all test results.
4. MUST securely generate and sign reports to not allow altering in any way.
5. MUST provide traceability to which version of the validated test plan was run.
6. MUST provide traceability to which version of the test scripts were run.

5.2 Test Data

Test Data is collected while executing the testing using a Certification Tool. A Certification Test is a collection of test cases which have the following data:

1. Each test case MUST include certification tool logging or explanation of the results.
2. Each test case SHOULD include a packet capture and relevant data.
3. Each test metric within a test case MUST indicate its associated pass/fail result.
4. Each test case MUST indicate a cumulative or overall pass/fail result which is a Logical AND of all test metrics.
5. Each test case MUST have a method for verifying logs and other test data are not altered in any way.

6 Certification Technical Review Committee

The Certification Technical Review Committee is a group of volunteers from prpl members responsible for reviewing Certification Requesters applications and test data. Once the technical components of the review is completed by the review committee it is sent to the prpl board for final approval before listing on the prpl Certification site.

6.1 Application Review process

The Certification Review Committee MUST approve the technical components of all applications for prpl Certification.

Applications will be assigned to a member of the committee with experience in the component of the application (prplOS, prplMesh, LCM, etc). The committee member performing the review will be selected by the Cert TWG Chair in an equal and fair manner to not overburden one member of the review committee. . The Cert TWG Chair MUST NOT assign a reviewer of the same company as the applicant. The Cert TWG Chair MUST NOT assign a reviewer that has a potential conflict of interest (eg direct competitor, ...).

The committee member will perform the following actions:

- Review a small sample of test data to confirm testing was executed appropriately.
- MUST verify the proper versions of the Certification Tool and Certification Tests were used in the test data.
- MUST confirm all the Core and Optional Features on the applications have been performed and submitted as part of the test data.
- MUST send the results of the review to the certification review committee mailing list allowing time for all the members to cross check all applications.
- MUST confirm that the testing was performed on the prplWare components as required for the certification.
    - Video, access to the component, or code inspection are all viable options for this step.

6.2 Committee Selection

The selection committee MUST be at least 4 members and SHOULD have at least 2 expert reviewers for each component of the certification program.

The membership SHOULD have the following members:

- One member from a test tool member
- One member from an operator.
- One member from a system Integrator
- One member from a silicon vendor

7 Program Lifecycle

7.1 Program Guide

The Program Guide has two version numbers <Major>.<Minor>. Minor is reserved for editorial, bug fixes, and clarifications. All other changes MUST update the Major number. Each revision MUST update the Revision History with a description of changes. When approved changes in this document take effect immediately.

7.2 Test Plans

The following describes the events in the evolution of the certification programs and the impact on members.

Each test plan has the following format for version numbers <Major>.<Minor>.<Patch>.

Condition	Impact
Patch Revision	When a patch number increases for a test plan it becomes available effective immediately. There is no need for a grace period as changes should be addressing issues.
Minor Revision	When a minor number increases for a test plan it becomes effective immediately as part of an active certification program. There will be a grace period of four weeks after a Minor Revision where previous version results as well as the new Minor Revision will be accepted.
Major Revision	When a major number increases for a test plan it becomes effective immediately as part of an active certification program. There will be a grace period of six months after a Major Revision where previous version results as well as the new Major Revision will be accepted.

7.3 Active Certification Programs

Active certification programs are defined as available test plans for use by certification requesters. When a grace period for a test plan expires it’s no longer considered an active certification program.