# Eclipse APERI Storage Management Project
# Integrated Development Verification Test Plan
# *-Phase 4-*

# Table of Contents
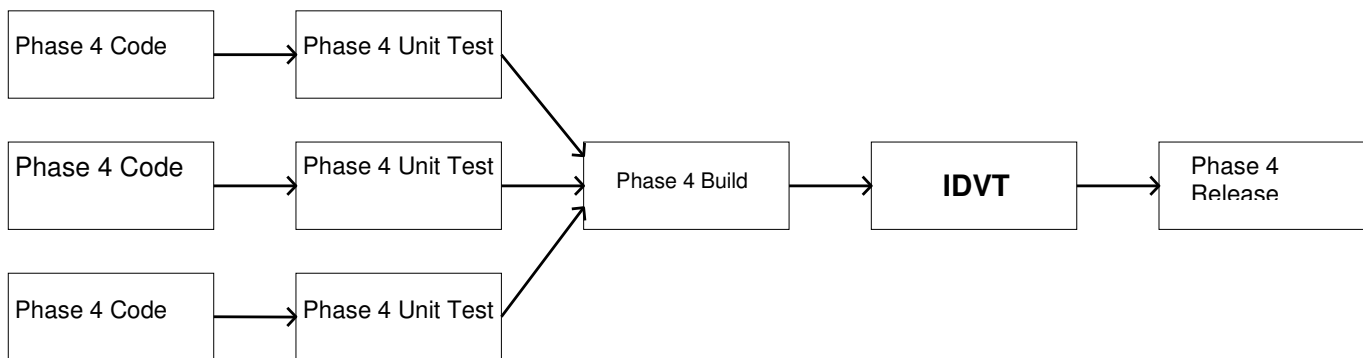
# 1. Introduction

This document specifies the Integrated Development Verification Test (IDVT) Plan for APERI, Phase 4. It identifies the items to be tested, the features to be tested, the types of testing to be performed, the personnel responsible for testing, the resources required to complete testing, and the risks associated with this plan.

This IDVT is performed by development after all unit testing is completed for Phase 4. IDVT concentrates on "good path" testing. Summarized test results are tracked via a document in the APERI External Development wiki Individual test cases are tracked by the test team using an external database, where individual test cases are defined in detail. A complete summary of the test cases are listed in this document.

IDVT for APERI is performed as follows:

| Phase 4 Code | → | Phase 4 Unit Test | | | |
|---|---|---|---|---|---|
| Phase 4 Code | → | Phase 4 Unit Test | → Phase 4 Build → | **IDVT** → | Phase 4 Release |
| Phase 4 Code | → | Phase 4 Unit Test | | | |

## 1.1 Objectives

The purpose of this IDVT is to test new features added to APERI since the phase 3 release and to verify that all phase three requirements are completed with only a limited number of minor bugs.

Testing includes, but is not limited to the following items:

- Europa Migration
- Aperi R0.4 GUI Installation feature
- Running the agent inside a Xen Virtual Machine
- IBM DS CIMOM Version 5.2.1 support.
- SAN Simulator R0.1 – Initial Contribution
- Viewing HTML pages with SWT browser widget

IDVT will test the Aperi Storage Manager as a complete application. This includes functions of the platform layer and application layer. IDVT will not directly test any of the extensibility or componentization product, vendors will eventually leverage to create features, plug-ins, or plug-in frames.

## 1.2 Scope

IDVT will cover new features implemented into the Aperi Storage Manager R0.4 (includes Eclipse Europa components) and its new GUI Installer. Some effort will also be made to test and benchmark Aperi's performance.

Phase 4 development utilizes the 5.2.1 CIM agent which is based on SMI-S 1.1; this work item is to plan for the future of Network Appliance SMI-S implementation. Novell added a new feature for running the data agent inside a virtual machine - for discovering paravirtualized block devices presented to guest Linux virtual machines via the XEN Hypervisor and XenBus. Finally, the SAN Simulator R0.1 – Initial Contribution will be shipped with Aperi R0.4.

## 1.3 IDVT Ecosystem Overview

For most test situations, the interim build is installed on existing acceptance test hardware, and IDVT test cases are performed in the acceptance test environment. In some cases, access to specialized hardware devices may require IDVT to be performed in a different site.

IDVT test cases are normally executed by members of the Aperi development team.

Defects found during IDVT will be logged in Bugzilla at Eclipse.

## 1.4 IDVT Entry Criteria

The entry criteria for Phase 4 of IDVT are:

- The IDVT Plan is approved by the Project Lead.
- The phase 4 user interface code is checked into the CVS repository.
- If special workarounds are required, they are specified in a handover document provided by the appropriate development area owner.
- The Phase 4 interim build can be installed through GUI installer for both Windows & Linux platforms
- The IDVT build passes an acceptance test (see below for details of the acceptance test).

Any deviation from these entry criteria requires the approval of the Project Lead.

Specific environmental requirements are identified below.

## 1.5 IDVT Exit Criteria

The IDVT exit criteria for this phase are:

- 100% of IDVT test cases attempted
- 95% of all IDVT test cases passed
- No 'blocked', 'critical' or 'major', severity defects, or viable workarounds provided for any 'blocked', critical' or 'major' severity defects.
- A "handover document" is produced to describe any manual procedures, workarounds, or limitations that the community needs to be aware of for this phase. This includes configuration instructions.

Any deviation from these exit criteria requires the approval of the Project Lead, and the Test Lead.

## 1.6 Domains of Testing

IDVT Phase 4 testing emphasizes SMI-S CIMOM agent support.  This interface will be used to test functions as the user would execute them. No special testing at the API level is performed during IDVT. It is expected that the nucleus of the IDVT Phase 4 test cases will be the CIMOM Version 5.2.1, Xen virtual machine & SAN Simulator supports.

In addition, install/configuration testing is performed as part of IDVT. IDVT phase 4 concentrates on the following scenario:

- All servers located on a single machine
- Windows platform
- Derby as repository
- Fresh install
- Limited number of tested hardware devices

Some IDVT phase 4 test cases specifically address the limitations of this scenario; that is, there will be some cases to test:

- Linux platform ( Xen virtual machine )
- Replacing the Derby repository with DB2

IDVT does not thoroughly test all supported hardware devices. Since IDVT concentrates on "good path" testing, any devices beyond the core set we concentrate on are not required to be tested by IDVT if they would exercise the same code paths as tested devices. But any hardware device that requires its own code path is tested during IDVT.

IDVT does not include publications, globalization, or accessibility.

# 2. Approach

## 2.1 Testing

### 2.1.1 Acceptance Testing

In order to insure that IDVT starts as scheduled, at least two comprehensive acceptance tests will be performed the week before IDVT is scheduled to begin.  Acceptance tests (smoke test) will be conducted to determine whether or not Aperi Storage Manager satisfies basic requirements.  Successful completion of the final test will allow for entry into the more detailed integration development verification testing.

A successful attempt of all of the following tasks constitutes a successful acceptance test:

1. Download and install the interim build (includes Eclipse Europa components)

2. Use of the gui installer in \install to install and configure the Data Server, Device Server, Report Server, Agents, and the GUI.

3. Verification that the Data Agent starts and is able to successfully register with the Data Server.

4. Verification that the Fabric Agent starts and is able to successfully register with the Device Server

5. Verification that the GUI comes up with no noticeable problems.

6. The initial agent probe works.

7. No errors appear in the Data Server log.

8. No errors appear in the Device Server log

9. No errors appear in the Report Server log

10. Verifying the add CIMOM agent works

11. Verifying the add Out Of Band Fabric agent works

12. Verifying the legacy SNMP discovery works.

13. Verifying the legacy CIMOM discovery works

14. Verifying the probe of Storage Subsystems works

15. Verifying default OS ping job executes

16. Verify the default OS probe job executes

17. Verifying discovered subsystems appear via the Topology Viewer

18. Verifying discovered volumes appear via the Topology Viewer

19. A random run through of  various reports (system reports, Disk Manager reports, Data Manager reports) to verify their accessibility.

## 2.1.2 Performance Testing

Some testing will be conducted to ensure that Aperi Storage Manager response time and scalability performs to expectations.

# 3. Environmental Requirements

This section lists both the necessary and desired properties required for the test environment.

### 3.1.1 Hardware

- One Windows server to host APERI server and agent components
- One Linux machine running APERI server and agent components
- Brocade switch with SMI agent
- McData switch with OPENconnectors SMI-S Interface
- Cisco SAN-OS CIM server
- DS8K, DS6K DS4K, ESS and SVC storage subsystems

### 3.1.2 Software

- DB2 to test as a replacement to derby

- Xen – a virtual machine software running on Linux distribution

### 3.1.3 Risks and Assumptions

- **Performance** - In some cases it will be difficult to create an environment required to emulate some of the desired performance tests.  For example, create volume in large configurations will not be possible due to the limited hardware available.  Efforts will be made to simulate environments for performance tests where the actual physical environment will not be possible to create.

# 4. Status Reporting

## 4.1 Status Reporting

Testing progress will be formally reported during the weekly Status Meetings.  The status will include all critical defects and all defects that impede the completion of IDVT.

A status report will be prepared by the test team to facilitate the meeting and inform management of IDVT status. This report will be posted on the Aperi external wiki by the end of each Friday until the exit of IDVT.  The report will contain the following information:

Test Cases

- The projected number of exposed test cases for each week in IDVT

- The projected number of successful test cases expected each week

- The actual number of exposed test cases each week

- The actual number of successful test cases each week

- The number of test cases being blocked due to defects

Defects

- The number of open defects

- The number of defects being worked on

- The number fixes to defects waiting to be verified

- The number of transferred defects

- The number of closed defects

- The number of returned defects

- The number of canceled defects

- The number of defects categorized by severity

- The number of open and working severity 1 and severity 2 defects

# 5. IDVT Test Cases

## 5.1  Phase 4

### 5.1.1  Required Environment

- See section **3.1.1** Hardware requirements

### 5.1.2  Required Staff

One IDVT tester.

### 5.1.3  Target Dates

- October 8, 2007 – start IDVT testing
- November 30, 2007 – end IDVT testing

### 5.1.4  Test Cases

#### 5.1.4.1  Viewing HTML pages with SWT browser widget

Source : Rodica
- Test on Windows w/IE6
- Test on Windows w/IE7
- Test on Windows w/FF2
- Test on SuSE Linux w/FF2/GTK
- Test on SuSE Linux w/Moz5/GTK
- Test on RedHat AS Linux w/FF2/GTK
- Test on RedHat AS Linux w/Moz5/GTK

#### 5.1.4.2  GUI Installation Improvement

- Install all components on windows platform using Derby as DB

- Install all components on windows platform using DB2 as DB
- Have agents reside on a different machine than the servers
- Install the AperiRcpGUI, by itself, on a separate machine than the servers
- Run Download third party components only using GUI install
- Install all components on Linux using Derby as db
- Install all components on Linux using DB2 as db

### 5.1.4.3  IBM DS CIMOM Version 5.2.1 support

- Test the CIM Agents Section under Configuration Utility Services
- Test the CIMOM Discovery Section under Configuration Utility Disk Manager tab
- Test the Storage Subsystems Section under Configuration Utility Disk Manager tab
- Test the Storage Subsystem Probes Section under Configuration Utility Disk Manager tab
- Disk Manager -> Verify Subsystems
- Disk Manager -> Verify Volumes
- Disk Manager -> Create Volumes
- Disk Manager -> Delete Volumes

### 5.1.4.4  SAN Simulator R0.1 – Initial Contribution

- DEVICE CIMOM SIMULATE - Snapshot based simulation
- DEVICE CIMOM SIMULATE - Configuration based simulation
- CORRELATE DEVICES - Switch to switch Connectivity
- CORRELATE DEVICES - Switch to Storage Connectivity

### 5.1.4.5  Running the agent inside a Xen Virtual Machine

Source : Robert Wipfel from Novell

 Robert had unit tested on SLES10, SLES10 sp1 and SLES10 sp1 Xen kernel (running
on physical and virtual machines).

- A final regression tests on the milestone build of Aperi R0.4 on the above platforms will be performed to satisfy the IDVT requirement.