

Persistence Subsystem Architecture Documentation

Version 1.9
<16/01/2016>

Content

1. Introduction	1
1.1 Purpose.....	1
1.2 Revision History	1
1.3 Abbreviations & Terminology.....	1
2. Persistence	3
2.1 Persistence as Black Box	3
2.1.1 Persistence Components in System Context.....	4
2.2 Persistence Components Overview	5
2.2.1 Persistence Client Library	5
2.2.2 Persistence Administration Service.....	6
2.2.3 Persistence Common Object	6
2.2.4 Persistence Health Monitor	7
3. Architecture Details	8
3.1 Persistence Client Library.....	8
3.2 Persistence Administration Service	8
3.3 Persistence Health Monitor	8
4. Appendix.....	9

List of Tables & Figures

Tables

Table 1 - Revision History 1
Table 2 – Abbreviations & Terminology 2

Figures

Figure 1 - Back Box View 3
Figure 2 - System Context 4
Figure 3 - Persistence Components Overview 5

1. Introduction

1.1 Purpose

The scope of this document covers the architecture of the Persistence Subsystem, including requirements, use cases and sequence diagrams.

1.2 Revision History

Revision	Change	Date
V 1.0	Setup of Document	25.11.2013
V 1.1	Added section about default data handling, the big picture section, the persistence failure section and the storage backends	14.01.2014
V 1.2	Added Notes to section Fehler! Verweisquelle konnte nicht gefunden werden. and Fehler! Verweisquelle konnte nicht gefunden werden.	14.02.2014
V 1.3	Updated section Fehler! Verweisquelle konnte nicht gefunden werden. Persistence Subsystem Overview" added section Fehler! Verweisquelle konnte nicht gefunden werden. "Cached Database Backend"	01.04.2014
V 1.4	Updated section Fehler! Verweisquelle konnte nicht gefunden werden. artitions.	09.05.2015
V 1.5	Added section Fehler! Verweisquelle konnte nicht gefunden werden. Black ox Use Cases.	28.08.2014
V 1.6	Added Requirement ID's to this documentation	11.09.2014
V 1.7	Added SQLite section under Guidelines, new template for documentation	09.10.2015
V 1.8	Switch to new template including complete rework or architecture documentation	30.10.2015
V 1.9	Changed document license to CC BY-SA 4.0	16.01.2016

Table 1 - Revision History

1.3 Abbreviations & Terminology

Abbreviation	Description
API	Application Programming Interface
DLT	Diagnostic Log and Trace
eMMC	embedded Multimedia Card
GENIVI	The GENIVI Alliance is a non-profit consortium whose goal is to establish a globally competitive, Linux-based operating system, middleware and platform for the automotive in-vehicle infotainment (IVI) industry.
IPC	Inter-Process Communication
Idbid	Logical Database Identifier
Linux VFS	Linux Virtual File System (Abstraction layer on top of a more concrete file system)
NAND	flash memory, a type of non-volatile computer memory

NOR	NOR Flash Memory, a type of non-volatile computer memory
NSM	Node State Manager (GENIVI lifecycle Component)
PAS	Persistence Administration Service
OSS	Open-Source-Software
RCT	Resource Configuration Table
PCL	Persistence Client Library
SSD	Solid-state drive
PCL	Persistence Client Library
PoC	Proof of Concept, see GENIVI definition: https://collab.genivi.org/wiki/display/genivi/SysArchComplianceBITTerms#SysArchComplianceBITTerms-ProofofConcept .
PFC	Persistence File Cache Component
PHM	Persistence Health Monitor
PCO	Persistence Common Object
ACID	Atomicity, Consistency, Isolation, Durability, see https://en.wikipedia.org/wiki/ACID

Table 2 – Abbreviations & Terminology

2. Persistence

The Persistence subsystem is responsible for handling persistent data. Persistent data is data that needs to be stored on a head unit between restarts, it includes all data read and modified during the lifetime of an infotainment system. Persistent data is stored in a non-volatile storage such as a hard disk drive or FLASH memory.

GENIVI defines the structure for the persistence management to meet standardization of GENIVI compliant platforms and some unique requirements introduced by the automotive domain.

The Persistence Management consists of the following components:

- Persistence Client Library
- Persistence Administration Service
- Persistence Common Object
- Persistence Health Monitor

There are a few reasons why GENIVI is developing this area, as opposed to simply saying that "many Linux storage solutions exist" and leaving it at that. First of all the overall goal is standardization of GENIVI compliant platforms so that applications running on GENIVI all use a common service for storing data. Secondly the automotive domain introduces some unique requirements:

- Strong robustness and reliability requirements from automotive OEMs for storing of critical data.
- Unreliable electrical environment, with possible power drop-out (engine cranking, blown fuses etc.)
- Lifetime requirements of automotive vehicle (in the order of 10-20 years) balanced against the lifetime of flash memory with its limited number of write (and to some extent read) cycles

The overall idea is to define a standardized common interface to persistent storage, under which system builders will still have significant flexibility in choosing the actual storage implementation that makes sense for each system. The persistence client library also provides an abstraction that avoids clients having to handle specific paths in the VFS and similar implementation details.

2.1 Persistence as Black Box

The following figure shows the interfaces of Persistence subsystem to the rest of the system. The provided interfaces which can be used by applications or others and the requested interfaces which have to be either fulfilled or stubbed by the product development.

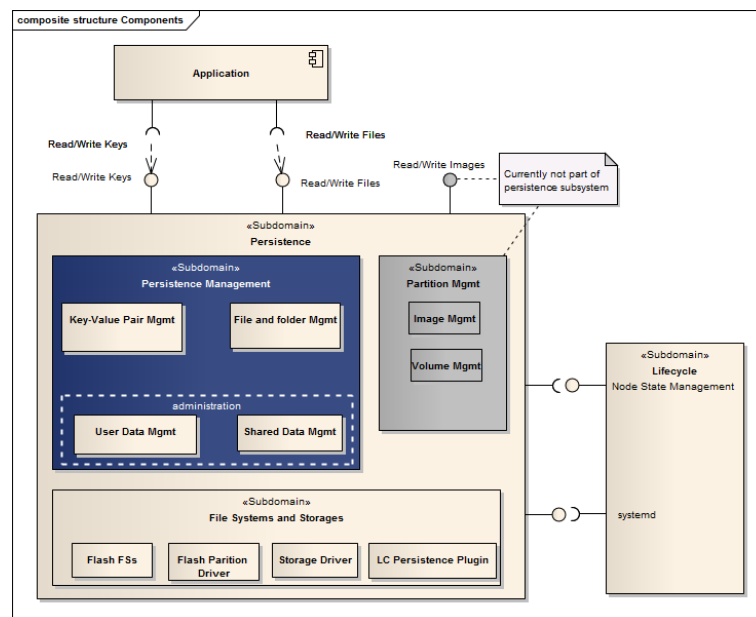


Figure 1 - Back Box View

2.1.1 Persistence Components in System Context

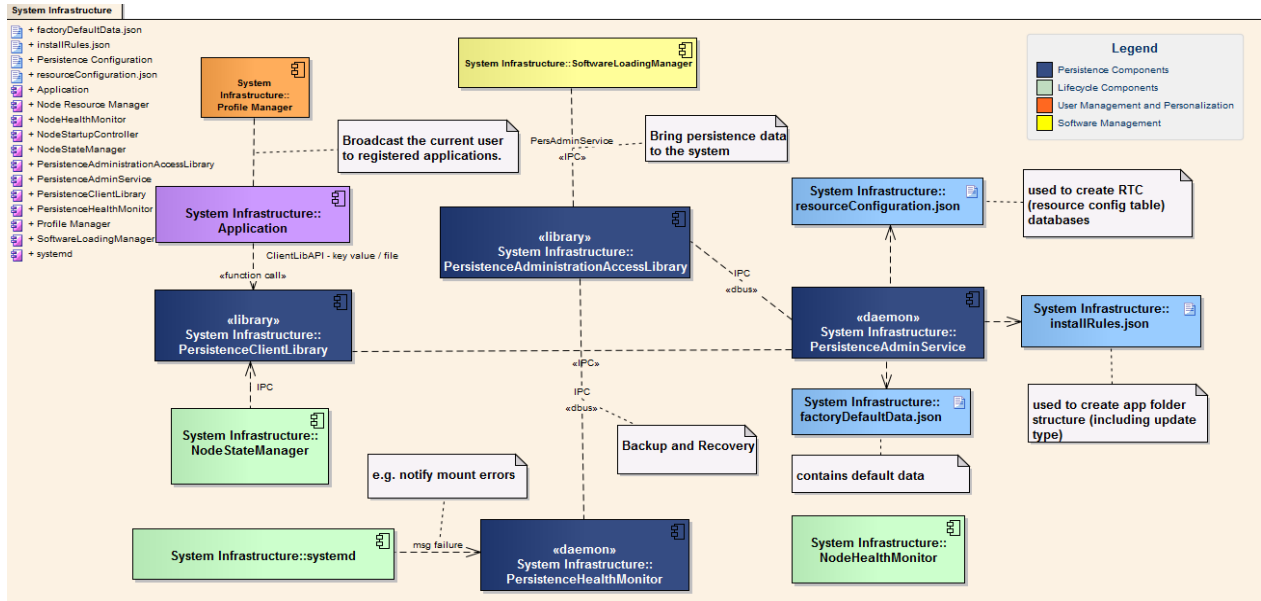


Figure 2 - System Context

2.2 Persistence Components Overview

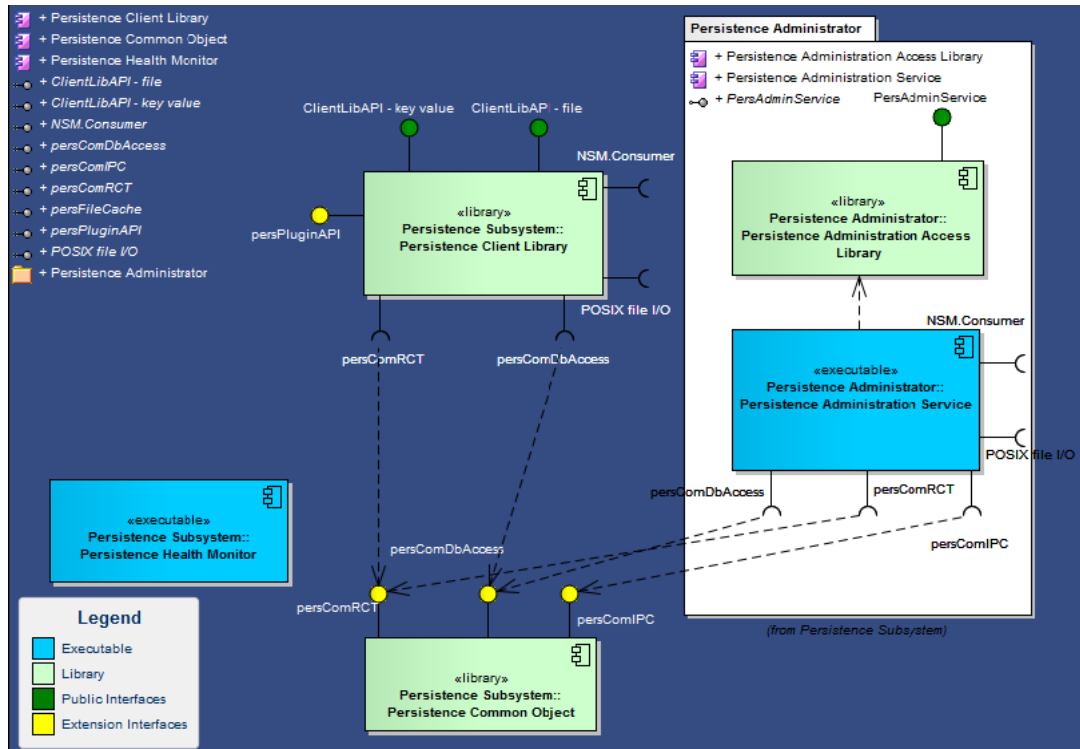


Figure 3 - Persistence Components Overview

2.2.1 Persistence Client Library

Responsibility:

- Provide an API to applications to read and write persistent data
- Provide a plugin API to allow users to extend the client library with custom storage solutions

GENIVI compliance status:

- Abstract P1

Repository:

- This component is implemented and available via GENIVI git repository.
- <http://git.projects.genivi.org/persistence/persistence-client-library.git>

Documentation:

- http://docs.projects.genivi.org/persistence-client-library/1.0/GENIVI_Persistence_Client_Library_UserManual.pdf

2.2.2 Persistence Administration Service

Responsibility:

- Installs the infrastructure for the data management
- Create default file and folder structure
- Setup access rights to the default file and folder structure
- Deploy default data
- Internal backup and restore database
- Provides a access library with a C-API (Persistence Administration Access Library)

GENIVI compliance status:

- Abstract P1

Repository:

- This component is available via GENIVI git repository.
- <http://git.projects.genivi.org/persistence/persistence-administrator.git>

Documentation:

- http://git.projects.genivi.org/?p=persistence/persistence-administrator.git;a=blob_plain;f=Administrator/doc/GENIVI_PersAdmin_ComponentSpecification.pdf

2.2.3 Persistence Common Object

Responsibility:

- Storage backend library used by PCL and PCS to access key/value and RCT data.
- Different storage mechanisms can be implemented, e.g.
 - Itzam/C
 - Key-value store
- IPC abstraction (planned)

GENIVI compliance status:

- Currently not in GENIVI compliance

Repository:

- This component is available via GENIVI git repository.
- <http://git.projects.genivi.org/persistence/persistence-common-object.git>

Documentation:

- http://docs.projects.genivi.org/persistence-client-library/1.0/GENIVI_Persistence_Common_Object_UserManual.pdf

2.2.4 Persistence Health Monitor

Responsibility:

- Implements recovery strategies in case of data corruption or persistence problems. The PHM uses the PAS to create backups and recover backups.
- Currently the architecture for this component is being developed
- Implementation will follow soon

GENIVI compliance status:

- Abstract P2

Repository:

- This component is available via GENIVI git repository.
- <http://git.projects.genivi.org/persistence/persistence-health-monitor.git>

Documentation:

- https://collab.genivi.org/wiki/download/attachments/1247269/GENIVI_Component_Specification_PersHealthMonitor.docx

3. Architecture Details

Details about requirements, black box use cases and sequence diagrams can be found in the corresponding GENIVI Component Specification.

3.1 Persistence Client Library

https://collab.genivi.org/wiki/download/attachments/1247269/GENIVI_Component_Specification_PersClientLibrary-2.3.1.docx

3.2 Persistence Administration Service

http://git.projects.genivi.org/?p=persistence/persistence-administrator.git;a=blob_plain;f=Administrator/doc/GENIVI_PersAdmin_ComponentSpecification.pdf

3.3 Persistence Health Monitor

https://collab.genivi.org/wiki/download/attachments/1247269/GENIVI_Component_Specification_PersHealthMonitor.docx

4. Appendix

Appendix 1 - Related Documents..... 9

Appendix 1: Related Documents

Document	Description	Version	Link
Persistence Client Library Component Specification	Component Specification	2.3.1	https://collab.genivi.org/wiki/download/attachments/1247269/GENIVI_Component_Specification_PersClientLibrary-2.3.1.docx
Persistence Administration Service Component Specification	Component Specification	1.0.0	http://git.projects.genivi.org/?p=persistence/persistence-administrator.git;a=blob_plain;f=Administrator/doc/GENIVI_PersAdmin_ComponentSpecification.pdf
Persistence Health Monitor Component Specification	Component Specification	1.0.0	https://collab.genivi.org/wiki/download/attachments/1247269/GENIVI_Component_Specification_PersHealthMonitor.docx
Persistence Client Library User Manual	User Manual	2.5	http://docs.projects.genivi.org/persistence-client-library/1.0/GENIVI_Persistence_Client_Library_UserManual.pdf
Persistence Common Object Library User Manual	User Manual	2.2	http://docs.projects.genivi.org/persistence-client-library/1.0/GENIVI_Persistence_Common_Object_UserManual.pdf

Appendix 1 - Related Documents

