# 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

<table>
<thead>
<tr>
<th>Revision</th>
<th>Change</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>V 1.0</td>
<td>Setup of Document</td>
<td>25.11.2013</td>
</tr>
<tr>
<td>V 1.1</td>
<td>Added section about default data handling, the big picture section, the persistence failure section and the storage backends</td>
<td>14.01.2014</td>
</tr>
<tr>
<td>V 1.2</td>
<td>Added Notes to section Fehler! Verweisquelle konnte nicht gefunden werden. and Fehler! Verweisquelle konnte nicht gefunden werden.</td>
<td>14.02.2014</td>
</tr>
<tr>
<td>V 1.4</td>
<td>Updated section Fehler! Verweisquelle konnte nicht gefunden werden. Artitions.</td>
<td>09.05.2015</td>
</tr>
<tr>
<td>V 1.5</td>
<td>Added section Fehler! Verweisquelle konnte nicht gefunden werden. Black ox Use Cases.</td>
<td>28.08.2014</td>
</tr>
<tr>
<td>V 1.6</td>
<td>Added Requirement ID's to this documentation</td>
<td>11.09.2014</td>
</tr>
<tr>
<td>V 1.7</td>
<td>Added SQLite section under Guidelines, new template for documentation</td>
<td>09.10.2015</td>
</tr>
<tr>
<td>V 1.8</td>
<td>Switch to new template including complete rework or architecture documentation</td>
<td>30.10.2015</td>
</tr>
<tr>
<td>V 1.9</td>
<td>Changed document license to CC BY-SA 4.0</td>
<td>16.01.2016</td>
</tr>
</tbody>
</table>

Table 1 - Revision History

1.3 Abbreviations & Terminology

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>DLT</td>
<td>Diagnostic Log and Trace</td>
</tr>
<tr>
<td>eMMC</td>
<td>embedded Multimedia Card</td>
</tr>
<tr>
<td>GENIVI</td>
<td>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.</td>
</tr>
<tr>
<td>IPC</td>
<td>Inter-Process Communication</td>
</tr>
<tr>
<td>ldbid</td>
<td>Logical Database Identifier</td>
</tr>
<tr>
<td>Linux VFS</td>
<td>Linux Virtual File System (Abstraction layer on top of a more concrete file system)</td>
</tr>
<tr>
<td>NAND</td>
<td>flash memory, a type of non-volatile computer memory</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>NOR</td>
<td>NOR Flash Memory, a type of non-volatile computer memory</td>
</tr>
<tr>
<td>NSM</td>
<td>Node State Manager (GENIVI lifecycle Component)</td>
</tr>
<tr>
<td>PAS</td>
<td>Persistence Administration Service</td>
</tr>
<tr>
<td>OSS</td>
<td>Open-Source-Software</td>
</tr>
<tr>
<td>RCT</td>
<td>Resource Configuration Table</td>
</tr>
<tr>
<td>PCL</td>
<td>Persistence Client Library</td>
</tr>
<tr>
<td>SSD</td>
<td>Solid-state drive</td>
</tr>
<tr>
<td>PCL</td>
<td>Persistence Client Library</td>
</tr>
<tr>
<td>PoC</td>
<td>Proof of Concept, see GENIVI definition: <a href="https://collab.genivi.org/wiki/display/genivi/SysArchComplianceBITTerms#SysArchComplianceBITTerms-ProofofConcept">https://collab.genivi.org/wiki/display/genivi/SysArchComplianceBITTerms#SysArchComplianceBITTerms-ProofofConcept</a></td>
</tr>
<tr>
<td>PFC</td>
<td>Persistence File Cache Component</td>
</tr>
<tr>
<td>PHM</td>
<td>Persistence Health Monitor</td>
</tr>
<tr>
<td>PCO</td>
<td>Persistence Common Object</td>
</tr>
</tbody>
</table>

*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 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

![Persistence Components Overview Diagram](image)

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](http://git.projects.genivi.org/persistence/persistence-client-library.git)

**Documentation:**
### 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](http://git.projects.genivi.org/persistence/persistence-administrator.git)

**Documentation:**

### 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)

**GENICI 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](http://git.projects.genivi.org/persistence/persistence-common-object.git)

**Documentation:**
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:
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


3.2 Persistence Administration Service


3.3 Persistence Health Monitor

Appendix 1: Related Documents

<table>
<thead>
<tr>
<th>Document</th>
<th>Description</th>
<th>Version</th>
<th>Link</th>
</tr>
</thead>
</table>