


<b>CIWS</b>		<b>CIWS – Customizable Instrument Workstation Software system for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	i

Internal Report IASF Bologna n. 638/2014

## The CIWS-FW Persistence

Prepared by: Name: V. Conforti Signature: \_\_\_\_\_ Date: 03/04/2014

Reviewed by: Name: A. Bulgarelli Signature: \_\_\_\_\_ Date: \_\_\_\_\_

Approved by: Name: M. Trifoglio Signature: \_\_\_\_\_ Date: \_\_\_\_\_




## DISTRIBUTION LIST

CIWS e-mail list	ciws@iasfbo.inaf.it




### DOCUMENT HISTORY

Version	Date	Modification
d0.1	03/04/2014	First draft

<b>CIWS</b>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	1

## TABLE OF CONTENTS

<b>1. INTRODUCTION .....</b>	<b>2</b>
1.1 PURPOSE OF THE DOCUMENT .....	2
1.2 DEFINITIONS, ACRONYMS AND ABBREVIATIONS .....	2
1.2.1 <i>Definitions and Terminology</i> .....	2
1.2.2 <i>Acronyms and abbreviations</i> .....	3
1.3 REFERENCES.....	3
1.3.1 <i>Applicable Documents</i> .....	3
1.3.2 <i>Reference documents</i> .....	3
1.4 OVERVIEW OF THE DOCUMENT .....	3
<b>2. THE PERSISTENCE IN THE CIWS-FW CONTEXT.....</b>	<b>4</b>
<b>3. PERSISTENCE ARCHITECTURE.....</b>	<b>6</b>
3.1 IW CONFIGURATION.....	7
3.1.1 <i>Measure Configuration</i> .....	7
3.1.2 <i>IW Data Acquisition</i> .....	7
3.1.3 <i>IW Data Retrieval</i> .....	7
<b>4. PERSISTENCE PROTOTYPE .....</b>	<b>8</b>
4.1 CONFIGURATION AND RETRIEVAL.....	8
4.2 PERSISTENCELIBC.....	9
4.3 PERSISTENCELIBCPP.....	10

<b>CIWS</b>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFB0-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	2

## 1. Introduction

### 1.1 Purpose of the document

The Customizable Instrument Workstation Software (CIWS) project is aimed at providing a software framework (CIWS-FW) for the development and operations of the Instrument Workstation (IW) required to support the Assembly, Integration, Verification and Testing (AIV/AIT) activities on scientific instruments for space borne experiments and ground-based telescopes in Astrophysics.


In addition, the CIWS-FW should facilitate the refurbishment of the IW software for the subsequent Commissioning and Operations phases to be carried out either in the mission Ground Segment of space-borne experiments, or in the Observatory site of ground-based telescopes.

The CIWS-FW architecture provides the Persistence Component which support the data acquisition operation through a database. The purpose of this document is to present the Persistence component and its interfaces with other CIWS-FW components.

### 1.2 Definitions, acronyms and abbreviations

#### 1.2.1 Definitions and Terminology

Name	Alias	Description
Acquisition pipeline	pipeline	a chain of processes that manage the data flow
Measure		It defines the process of setting up the instrument, followed by the acquisition of at least one detector and the storage of the obtained data product in mass memory and/or in main memory for display.
Measure ID		An incremental number that univocally identifies each measure.
Measure Plan	Measure	A sequence of consecutive and correlated Measure sessions.
Measure Plan ID	Pid	Measure plan identifier.
Measure Session		A sequence of correlated Measure, (e.g.: the smallest schedulable unit containing all information necessary to execute sequentially and without interruption a set of correlated measures, involving a single telescope preset).
Measure Session ID	Sid	Measure session identifier
Measure Block		The session can be split in more blocks when it is required.
Measure Block ID	Bid	Measure Block identifier
IDLE mode		This basic mode is a generalization of the various operating modes that do not generate detector data. Only instrument functions necessary for operating safely are powered on and the instrument is either being configured (Instrument Preset) or is ready for measure.

<b>CIWS</b>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	3

RUN mode		This basic mode is a generalization of the various operating modes that generate detector data
----------	--	--

## 1.2.2 Acronyms and abbreviations

IW	Instrument Workstation
API	Application Programming Interface
CIWS	Customizable Instrument Workstation Software


## 1.3 References

### 1.3.1 Applicable Documents

### 1.3.2 Reference documents

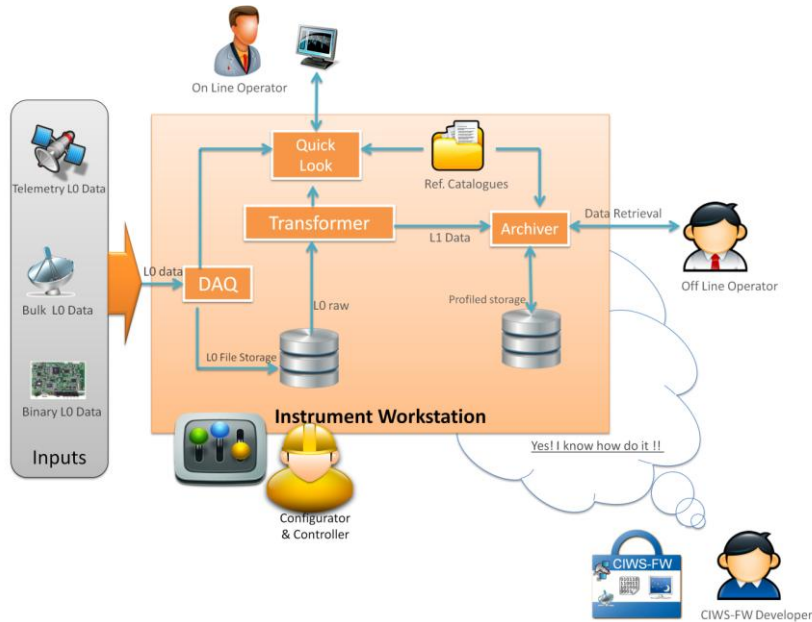
- RD [1] CIWS User Requirement Document (URD) - M. Trifoglio - 16/09/2013.
- RD [2] CIWS Software Specification Document (SSD) - A. Bulgarelli - 19/06/2013.
- RD [3] CIWS-FW: a Customizable InstrumentWorkstation Software Framework for instrument-independent data handling" In proceeding of ADASS XIII Conference Waikoloa (HI) October 2013 - V. Conforti in behalf of CIWS

## 1.4 Overview of the document

<h1>CIWS</h1>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>				
	Code: <b>CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page: 4

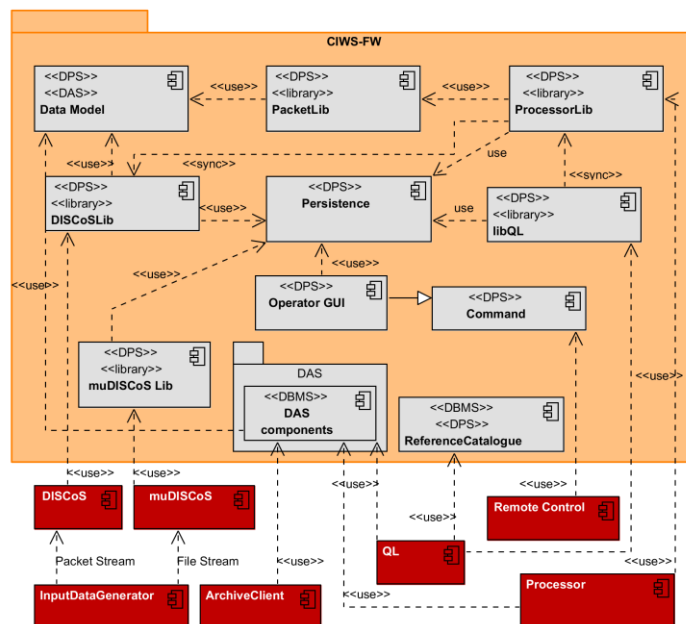
## 2. The Persistence in the CIWS-FW context

The CIWS-FW concept detailed in presented in the RD [3] is depicted in figure 2-1.




2-1 CIWS-FW Model View

The CIWS-FW provides the tools for the IW building. the IW allows the near real time acquisition of input packet data streams and file data streams that should be transformed and archived according to the data models defined by the user. The IW shall provides configuration and controller GUI to control and monitor the operations. The CIWS-FW provide as Configurator & Controller GUI the Control Panel.



2-2 CIWS-FW Architecture

<b>CIWS</b>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	5

The figure 2-2 depicts the CIWS-FW architecture. The Persistence component is actually prototyped for provide services to the following CIWS-FW components:

- Operator GUI
- DISCoSLib
- ProcessorLib

The Operator GUI uses the Persistence components facilities to load the configuration of the IW and to set the acquisition plan configuration;

The DISCoSLib has in charge the packets stream acquisition and storage. Before to go in RUN mode the DISCoSLib load the configuration through the Persistence in order to couple the data acquired to the measure plan. Furthermore the DISCoSLib notify to the Persistence the acquisition status and the creation files.

The ProcessorLib has in charge the data level transformation, and it uses the Persistence to notify the creation files.

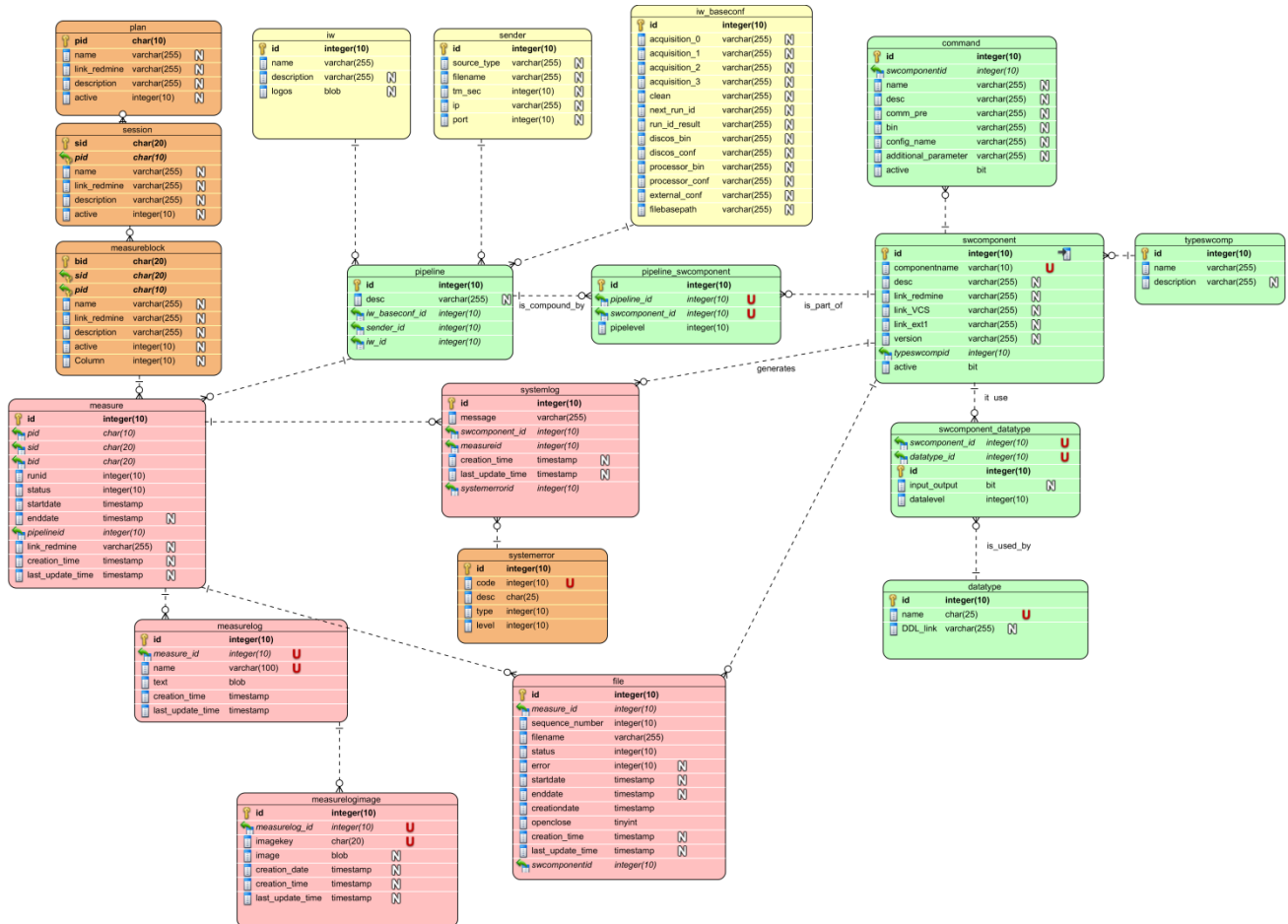
The Persistence also provide a system log and system error notification functions for the DISCoSLib and ProcessorLib.





### 3. Persistence Architecture

The Persistence provide softwares which sharing the same database. The database keeps track of the IW status and the acquisition status. The figure 3-1 depicts the Entity Relation Ship diagram.




3-1 Entity Relation Ship Diagram

The database support the following services:

1. IW Configuration;
2. Measure configuration
3. Data Acquisition;
4. Data Retrieval.

The IW configuration and data retrieval tasks are supported by a web interface of Persistence. The Measure configuration is available inside the Control Panel which has direct access to the database. The PersistenceLib shall be used in the data acquisition software. The PersistenceLib is available in the following two releases:

- PersistenceLibC (to be used in C programming language software component);
- PersistenceLibCpp (to be used in C++ programming language software component);

<b>CIWS</b>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	7

### 3.1 IW Configuration

During the IW development a Persistence web provide the following services:

- sw components definition;
  - sw component commands;
  - data type;
- pipelines definition;
- error codes definition;

#### 3.1.1 Measure Configuration

After the implementation of the IW software the Control Panel allows the operator to configure the measure plan. This configuration is the insert of the plan session and block in the database as active configuration.


#### 3.1.2 IW Data Acquisition

The software components while execute the data acquisition tasks must use the PersistenceLib methods to notify:

- Open and Close of the measures;
- Open and Close of the files;
- Logs;
- Errors;

#### 3.1.3 IW Data Retrieval

The data retrieval functions are always available through a web interface. It allows the user to retrieve all information inserted in the database.

<h1>CIWS</h1>		<b>Customizable Instrument Workstation Software (CIWS) for telescope-independent L0/L1 data handling</b>					
	<b>Code: CIWS-IASFBO-TN-007</b>	Issue:	0.1	DATE	<b>03-04-2014</b>	Page:	8

## 4. Persistence Prototype

The Persistence prototype provide the web interface for the IW configuration, measure configuration, and data retrieval. Also the libraries PersistenceLibC and PersistenceLibCpp are developed to support the acquisition state notifications.

### 4.1 Configuration and Retrieval

Figure 4-1 and 4-2 show the user interface to configure the IW: pipelines, software components, data types.



4-1 Web Interface for the IW Configuration a

**CONTROL PANEL - IW CONFIGURATION**

[Measure Plan](#) - [IW](#) - [Pipelines](#) - [Swcomponents](#) - [Data types](#)

**Actions**

New Swcomponent

**Type Swcomponents**

List Typeswcomps

**Error Codes**

List System Errors Code

New System Error Code

**Commands**

List Commands

New Command

**Pipelines**

List Pipelines

New Pipeline

New Pipeline <-> Swcomponent

**Data Types**

List Datatypes

New Datatype

### Swcomponents

Id	Componentname	Desc	Link Redmine	Link VCS	Link Ext1	Version	Typeswcompid	Active	Actions
1	ASTRI DAQ	Packet by Packet daq sw					<a href="#">DAQ</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
6	S21Proc	desrc					<a href="#">Processor</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
8	DISCoS						<a href="#">DISCOS</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
9	Monitor						<a href="#">DISCOS</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
10	Clean						<a href="#">DISCOS</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
11	Sender						<a href="#">DISCOS</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
12	MCS	DAQ file by file					<a href="#">DAQ</a>		<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
13	S21_DAS_pr	DAS processor					<a href="#">Processor</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>
14	TEST						<a href="#">Analysis</a>	1	<a href="#">View</a> <a href="#">Edit</a> <a href="#">Delete</a>

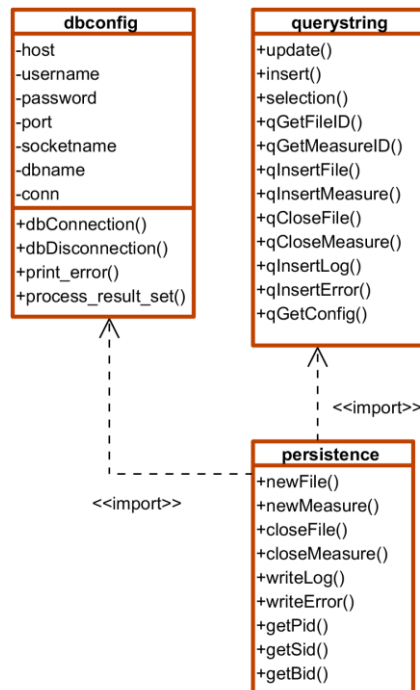
Page 1 of 1, showing 9 records out of 9 total, starting on record 1, ending on 9

[< previous](#)
[next >](#)

*4-2 Web Interface for the IW Configuration b*

## 4.2 PersistenceLibC

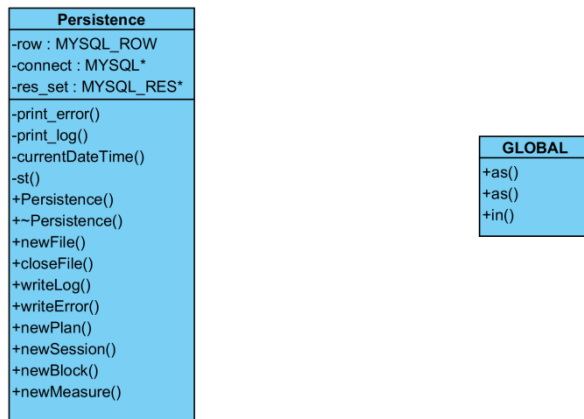
The PersistenceLibC library is developed in C language and it is available in the CIWS-FW. The figure 4-3 shows a resume of functions offered by the library.



*4-3 PersistenceLibC functions diagram*

### 4.3 PersistenceLibCpp

The PersistenceLibCpp library is developed in C++ language and it is available in the CIWS-FW. The figure 4-4 report the PersistenceLibCpp class diagram.



4-4 PersistenceLibCpp Class diagram