IDM UID
34SDZ5

VERSION CREATED ON / VERSION / STATUS
11 Mar 2024 / 7.3 / Approved

EXTERNAL REFERENCE / VERSION

Rules or Handbooks or Guidelines

CODAC Core System Overview

This document is an overview of the CODAC Core System software distribution. It is a part of the CODAC Core System documentation as well as a satellite document of the Plant Control Design Handbook (PCDH).

Approval Process			
	Name	*Action*	*Affiliation*
Author	**Lange R.**	**11 Mar 2024:signed**	**IO/DG/CP/CIC/DCS**
Co-Authors			
Reviewers	**Park M.**	**20 Mar 2024:recommended**	**IO/DG/CP/CIC/DCS**
Approver	**Wallander A.**	**22 Mar 2024:approved**	**IO/DG/CP/CIC**
Document Security: Non-public - Unclassified *RO: Stepanov Denis*			
Read Access	**AD: ITER, AD: External Collaborators, AD: IO_Director-General, AD: External Management Advisory Board, AD: OBS - Controls & Int. Commissioning Program (CIC) - EXT, AD: IDM_Controller, AD: OBS - Data, Connectivity and Software Project (DCS) - EXT, AD: OBS - Data, Connectivity and Software Project (DC...**		

Change Log

CODAC Core System Overview (34SDZ5)

Version	***Latest Status***	***Issue Date***	***Description of Change***
v1.0	In Work	11 Dec 2009	
v1.1	In Work	13 Jan 2010	No changes, just PDF version
v1.2	Signed	05 Feb 2010	Version for review.
v1.3	Signed	10 Feb 2010	Updated from comments on vs 1.2
v1.4	Approved	12 Feb 2010	Minor editions + added last section on user doc.
v1.5	Approved	19 Jul 2010	Minor changes to cope with the exstence of vs 1.1.
v2.0	In Work	24 Sep 2010	1st draft for version2.0.
v2.1	In Work	30 Nov 2010	2nd draft for inclusion into 2.0b3.
v2.2	Signed	16 Dec 2010	Develop the features, integrate internal comments. For beta4.
v2.3	Signed	07 Jan 2011	Version for review. Includes changes from J.Poole.
v2.4	Signed	24 Jan 2011	Updated according to the final features of release 2.0. Updated the Self-description chapter (chapter 4), add illustrations. Suppress the empty CODAC IDE section (omponent has been de-scoped) Added description of System Monitoring (section 6.2) Updated the ICH sample section (section 7.1) Updated the PLC sample section (section 7.2) Added the plant system I&C health monitoring (section 7.3). Updated all figures from latest code.
v2.5	Approved	10 Feb 2011	Final for 2.0.0 & PCDH 6.1. Added PCDH satellite document section and SDN/TCN API.
v2.6	Approved	12 Jul 2011	Change the recommended resolution for displays from 1400x1050 to 1280x1024. Include the 2.1 release and update the list of documents acording to the changes in the 2.1 release.
v3.0	Approved	14 Feb 2012	Updated for vs3.0: 1) add the 3.0 version in history (section 1.5) 2) mention system profiles (section 2.2) 3) mention shared repository for CSS contributions (section 2.3) 4) Update RHEL version and other component versions (chapter 3) 5) add new SDD tools (parser sync) and develop SDD web applciation (Chapter 4) 6) mention new project structure and CBS (chapter 4) 7) add icon libary for BOY (section 5.3) 8) add C/C++ and Java to Python as programming languages (section 5.6.2) 9) extend VDCT usage with SDD parser (section 5.6.3) 10) add fast cntroller sample (section 7.2) 11) new text on future releases (chapter 8) 12) list of documents updated In addition: - Changes to make the document more version neutral - Suppression of some details (ex: RHEL licences type, user groups) - Suppression of references to version 1 and to version 1 components (EDM qualified deprecated).
v3.1	Approved	17 Jun 2012	Changes for the 3.1 release + alignment with other documents. Add version 3.1 in history Remove PC specifications from this document while the installation manual should be the reference. Rename "Mini-CODAC"chapter to "CODAC Services" according to the

			CODAC DDD terminology. Separate development tools in a dedicated chapter Add a section on EPICS IOC development Add a section ITER Maven commands, insert new SVN commands. Remove the "Future releases" chapter. Information on new features and future releases shall be in the release notes and roadmap. Add the digital IO board as in 3.1. Update the list of documents. Update fig 1 with CODAC DDD schema (with DAN).
v4.0	Approved	08 Feb 2013	Updated for version 4.0. Additions are marked with a bar: cubiclle monitoring, COS support, SDN API, example suppression... Reworded to reflect that from version 4.0, version 2.0 is not anymore.supported.
v4.1	Approved	03 Jul 2013	Updated for the release of CODAC Core System vs 4.1 (July 2013). Addition of the 4.1 version in the History section (section 1.5) Addition of the handling of SDN data and SDN programs in SDD (section 4 and 7.5) Addition of web browser interface for alarms (section 5.4) Changes in version 4.0 and 4.1 and indicated with a left margin bar.
v4.2	Approved	10 Feb 2014	Minor changes for the CCS 4.2.0 release: - Update figure 1 with CODAC PDR version - Add version 4.2 in history - Add list of additional CSS tools (section 5.6) - Add support for NI FlexRIO (section 7.4) - Add section on DAN support (section 7.7) - Add SPSS and NI RIO documents in the list of documentation (chapter 8)
v4.3	Approved	26 Jun 2014	Minor update for the 4.3.0 release: - Add version 4.3 in history - Expand Maven editor (figure) - Removed VDCT - Confirm & clarify the 1 micor-second accuracy for system time with TCNd - Added PXIe-63868 support - Update list of document (chapter 8)
v5.0	Approved	09 Feb 2015	Updated for CCS 5.0 release - Announce the change for support (reducing 3.x support on 5.0.0 release). - Update history, DAN description, hardware list and documentation table. - Update the release plan by replacing major/minor with A/B. - Remove history in features.
v5.1	Approved	09 Jul 2015	Update to align with CCS 5.1 and other changes since the CCS 5.0 release: 1) Add Plant Other Controller (POC) in acronyms (section 1.4) 2) Add the CCS 5.1 release with main changes in the history (section 1.5) 3) Suppress the section "Public Users" in chapter 2 (distribution suppressed) 4) Replace quarters with months for the planned dates of releases A&B (section 2.4) 5) Add Install/remove commands in the list for ITER Maven tools (section 6.2) 6) Add "Support for Programs" section (7.8) 9) Update the documentation list with the new documents
v5.2	Approved	28 Jan 2016	Improvements from the review of the last version: update of references, missing acronyms, more overview (chapter 2), structure improved,screen dump updated, etc. Updates for the CCS 5.2.0 release: 5.2 in history (section 1.5), release B may be a maintenance version (section 2.2), SDD web application role reduced (chapter 5), operational applications added (section 6.1), documentation list updated (chapter 9)

v5.3	Approved	14 Jun 2016	Update for CCS 5.3.0 release - Add new version in history (section 1.6) - Announce that only one version / year from 2017 (section 2.2) - Add 2 documents in chapter 9 (SDN Archiver & installation manager) + Minor text improvements to adjust with scope changes.
v5.4	Approved	08 Feb 2017	Update for CCS 5.4.0 release: - add release of CCS 5.4.0 in section 1.6 (history) - add list of device support in section 4.2 (EPICS) - limited support for PXI-6682 in section 8.3 (Timing & synchronization) - update documentation list in chapter 9
v6.0	Signed	13 Feb 2018	Update for the release of CCS 6.0 1) add the new release in the history (section 1.6) 2) clarify the release scheme (section 2.2) 3) announce support change for older versions (section 3.3) 4) suppress SDD web application (chapter 5) 5) add PXIe-6363 support (section 8.4) 6) update list of documents (chapter 9)
v6.1	Approved	15 Feb 2018	Same changes as in previous version for CCS 6.0: Update for the release of CCS 6.0 1) add the new release in the history (section 1.6) 2) clarify the release scheme (section 2.2) 3) announce support change for older versions (section 3.3) 4) suppress SDD web application (chapter 5) 5) add PXIe-6363 support (section 8.4) 6) update list of documents (chapter 9) and also: 7) update RHEL version (section 4.1) 8) update EPICS version (section 4.2)
v6.2	Approved	27 Feb 2019	- Text improvements at various places + some icons - Add release of 6.1 in history (Mar-2019) - Merge the SDD chapter into a general chapter on development tools (chapter 5) - Add the workflow presentation in this chapter (section 5.3) - Add SVN in this chapter (section 5.5) - Complete the set of SDD data components - Update CS-Studio presentation and illustration, add e-log - Mention UDA in the 2 archiving sections (CS-Studio & DAN) - Extend the scope of Chapter 7 to all supported controllers, including PSH. - Develop the PLC section to cover development in PLCs and SPSS (section 7.2) - Add a section about program (7.8)
v6.3	Approved	09 Oct 2020	- update and improve wording in many places - update reference documents (new URLs, new versions) - add version 6.2 to history - update figure 2-1 (physical architecture) - update description of release scheme (2.2) - update EPICS logo (4.2) - update figure 5-2 (development workflow) - add OPC UA integration for PLC at the same level as SPSS based integration - update list of available user documentation
v7.0	Approved	08 Mar 2022	Update for CCSv7.0: - improve wording in many places - add 6.3 and 7.0 to version history - update the release scheme description (2.2) - update version scheme for user support (3.3) - updates for RHEL 8.5 (4.1) - update EPICS module list (4.2)

			- update I/O board list (7.4) - update documentation list (8).
v7.1	Approved	20 Mar 2023	Minor update for CCS v7.1: - cleaning up text - update/review of acronyms - update/review of references
v7.2	In Work	11 Mar 2024	Update for CCSv7.2: - CCS History - acronyms - add RTF and NDSv3 to supported features - list of documents
v7.3	Approved	11 Mar 2024	Regenerated TOC

Table of Contents

1 INTRODUCTION

1.1 Purpose

This document is a public document to provide ITER users with an overview of the software distribution named *CODAC Core System* (CCS) which is distributed to the suppliers responsible for the development and delivery of the Instrumentation and Control (I&C) for the ITER plant systems.

This document serves also a satellite document for the ITER Plant Control Design Handbook (PCDH) [RD1] as a reference for the I&C software environment.

1.2 PCDH Context

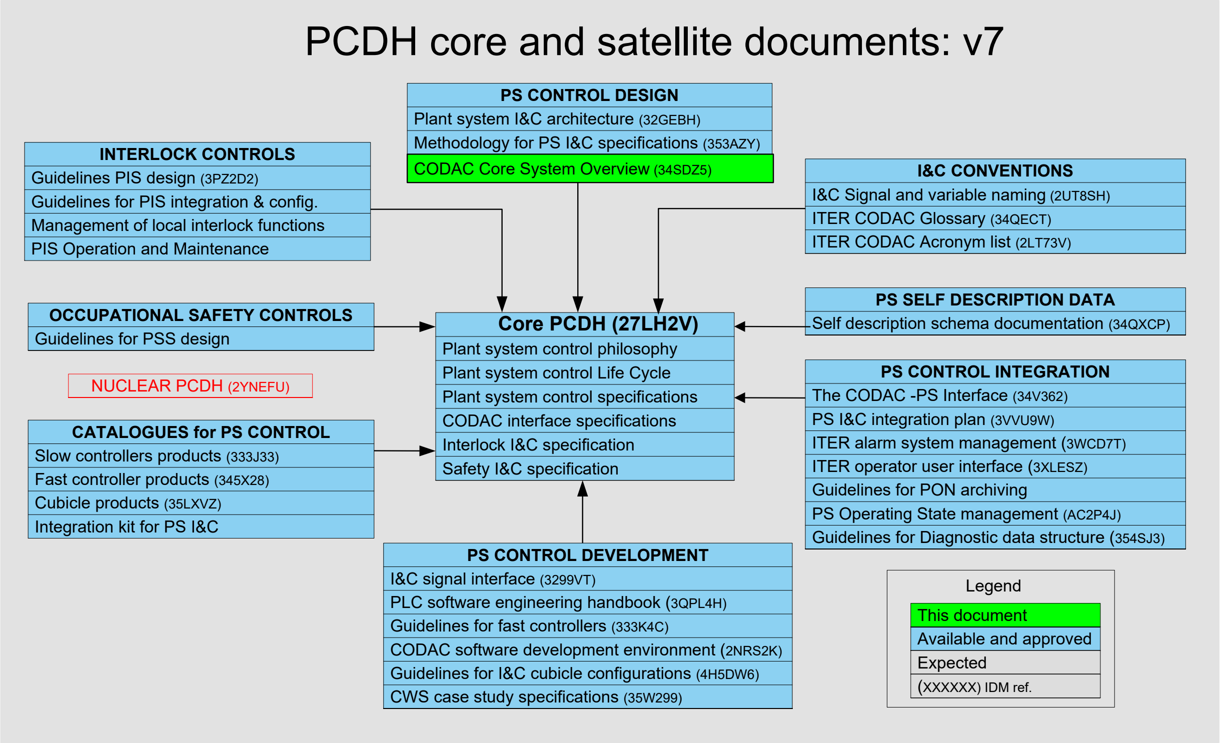

PCDH core and satellite documents: v7
PS CONTROL DESIGN
Plant system I&C architecture (32GEBH)
Methodology for PS I&C specifications (353AZY)
CODAC Core System Overview (34SDZ5)
INTERLOCK CONTROLS
Guidelines PIS design (3PZ2D2)
Guidelines for PIS integration & config.
Management of local interlock functions
PIS Operation and Maintenance
I&C CONVENTIONS
I&C Signal and variable naming (2UT8SH)
ITER CODAC Glossary (34QECT)
ITER CODAC Acronym list (2LT73V)
OCCUPATIONAL SAFETY CONTROLS
Guidelines for PSS design
NUCLEAR PCDH (2YNEFU)
Core PCDH (27LH2V)
Plant system control philosophy
Plant system control Life Cycle
Plant system control specifications
CODAC interface specifications
Interlock I&C specification
Safety I&C specification
PS SELF DESCRIPTION DATA
Self description schema documentation (34QXCP)
PS CONTROL INTEGRATION
The CODAC -PS Interface (34V362)
PS I&C integration plan (3VVU9W)
ITER alarm system management (3WCD7T)
ITER operator user interface (3XLESZ)
Guidelines for PON archiving
PS Operating State management (AC2P4J)
Guidelines for Diagnostic data structure (354SJ3)
CATALOGUES for PS CONTROL
Slow controllers products (333J33)
Fast controller products (345X28)
Cubicle products (35LXVZ)
Integration kit for PS I&C
PS CONTROL DEVELOPMENT
I&C signal interface (3299VT)
PLC software engineering handbook (3QPL4H)
Guidelines for fast controllers (333K4C)
CODAC software development environment (2NRS2K)
Guidelines for I&C cubicle configurations (4H5DW6)
CWS case study specifications (35W299)
Legend
This document
Available and approved
Expected
(XXXXXX) IDM ref.

Figure 1-1 – PCDH Documentation Structure

The Plant Control Design Handbook [RD1] defines methodology, standards, specifications and interfaces applicable to the ITER Plant Systems instrumentation and control system life cycle. I&C standards are essential for ITER to:

- Integrate all plant system I&Cs into one integrated control system;
- Maintain all plant system I&Cs after delivery acceptance;
- Contain cost by economy of scale.

At the heart of the PCDH, a core document presents the plant system I&C life cycle and recaps the main rules to be applied to plant system I&Cs for conventional controls, interlocks and safety controls. Some I&C topics will be explained in greater detail in dedicated documents associated with the PCDH, as presented in Figure 1-1. This document is one of them.

1.3 Structure of this Document

This document is structured as follows:

- Chapter 1 describes the purpose of this document and its context with regards to the Plant Control Design Handbook. It also lists references, defines abbreviations and describes the history of how this document has evolved.
- Chapter 2 gives an overview of the CODAC Core System as well as its release scheme.
- Chapter 3 describes how to become a registered user, how to obtain the CODAC Core System distribution and how to get support.
- Chapter 4 lists the system components of the CODAC Core System.
- Chapter 5 describes development tools that come packaged in the CODAC Core System.
- Chapter 6 describes the CODAC services, mostly for operators, that it implements.
- Chapter 7 provides information about the tools for controllers.

This document intends to serve as an introduction to the CODAC Core System. More details are contained in the documentation listed in chapter 8.

1.4 Reference Documents

[RD1] Plant Control Design Handbook (ITER_D_27LH2V)
https://www.iter.org/mach/codac/PlantControlHandbook

[RD2] EPICS – Experimental Physics and Industrial Control System
https://epics-controls.org/

[RD3] ITER CODAC Acronyms List (ITER_D_2LT73V)
https://portal.iter.org/departments/CHD/CODAC/Pages/Acronyms.aspx

[RD4] CODAC Core System public web page:
https://www.iter.org/mach/codac/CoreSystem

[RD5] On-line Learning Centre – CODAC Core System Training
https://sharepoint.iter.org/support/learning/SitePages/CODAC%20Core%20System%20Training.aspx

[RD6] CODAC DDD (ITER_D_6M58M9)

[RD7] ITER catalogue for I&C products – Slow controllers PLC (ITER_D_333J63)

[RD8] ITER Catalog of I&C products – Fast Controllers (ITER_D_345X28)

1.5 Acronyms

For a complete list of ITER CODAC abbreviations, please refer to [RD3].

API	Application Programming Interface
CA	Channel Access (EPICS network protocol)
CBS	Control Breakdown Structure
CCS	CODAC Core System

CODAC	Control, Data Access and Communication
COS	Common Operating State
cRIO	Compact Reconfigurable Input Output
CS-Studio	Control System Studio
DAN	Data Archiving Network
DDD	Design Description Document
DNF	Dandified YUM
EPICS	Experimental Physics and Industrial Control System
GUI	Graphic User Interface
HMI	Human-Machine Interface
HPN	High Performance Network
I&C	Instrumentation and Control
I/O	Input / Output
IDE	Integrated Development Environment
IO	ITER Organization
MRG-R	Messaging Realtime Grid - Realtime
NI	National Instruments
NTP	Network Time Protocol
OPC UA	Open Platform Communications Unified Architecture
OPI	Operator Interface
PA	Procurement Arrangement
PCF	Plant Controller, Fast, or Fast Controller
PCI	Peripheral Component Interconnect
PCIe	PCI Express
PCDH	Plant Control Design Handbook
PLC	Programmable Logic Controller or Slow Controller
POC	Plant Other Controller
PON	Plant Operation Network
PSH	Plant System Host
PSOS	Plant System Operation State machine
PTP	Precise Time Protocol
PV	Process Variable
PVA	PVAccess (EPICS network protocol)
PXI	PCI eXtensions for Instrumentation

RHEL	Red Hat Enterprise Linux
RIO	Reconfigurable Input / Output
RO	Responsible Officer
RPM	RPM Package Manager (formerly "RedHat Package Manger")
SDD	Self-description Data
SDN	Synchronous Databus Network
SNL	State Notation Language
SPSS	Standard PLC Software Structure
SVN	Subversion
TCN	Time Communication Network
TCR	Temporary Control Room
UDA	Unified Data Access
URL	Uniform Resource Locator
YUM	Yellowdog Updater, Modified

1.6 Short History of CODAC Core System

- Feb-2010 – Version 1.0, preliminary release with first PLC integration and SDD tools.
- Jul-2010 – Version 1.1, improvements of packaging and SDD components.
- Feb-2011 – Version 2.0, stabilized release for developers, introducing the new environment for SDD and mini-CODAC, stable data model and PLC interface.
- Jul-2011 – Version 2.1, minor release to extend the support for fast controllers, to update the CS-Studio toolkit (BOY) and to distribute bug fixes.
- Feb-2012 – Version 3.0, consolidated release for I&C development: RHEL vs 6, project structure, updated components (ITER and external).
- Jun-2012 – Version 3.1, minor release with new features in CS-Studio tools, SDD tools and Maven commands and with the support of a new digital I/O board.
- Feb-2013 – Version 4.0, improvements in CS-Studio tools, SDD tools and Maven commands, addition of the SDN API, support for Common Operating State (COS) and support for redundant PLC.
- Jul-2013 – Version 4.1, improvements in CS-Studio tools, SDD tools and Maven commands. Support for SDN configuration and redundant PLC in SDD. New version of the timing software with support for new boards.
- Feb-2014 – Version 4.2, addition of CS-Studio tools, extension of Maven commands, improvement of existing CS-Studio and SDD tools, support for FlexRIO, draft DAN API.
- Jul-2014 – Version 4.3, extensions of the CS-Studio, SDD and Maven tools, support for the PXIe-6368 board, new version of the TCN library and TCNd, and improvements of the SDN and DAN support.

- Feb-2015 – Version 5.0, new EPICS version (3.15.1), upgrade of the OS (RHEL 6.5, MRG-R 2.5), extensions of SDD, Maven and CS-Studio tools, new driver for PLCs, support for CompactRIO, improvements in TCN,SDN and DAN support.
- Jul-2015 – Version 5.1, addition of the support for Plant Other Controllers (POC), of events in PLC communications and of EPICS interface for the N.I. RIO boards; extensions for OPIs and for the PXIe-6368 board support; corrections and enhancements in all but very few components.
- Feb-2016 – Version 5.2, addition of operational applications for FAT/SAT, corrections and enhancements in the existing components.
- Jul-2016 - Version 5.3, CBS4/CBS5 support, support for huge projects, corrections and enhancements in the existing components
- Feb-2017 - Version 5.4, extension of EPICS device support, improved PTPd support implementation, corrections and enhancements in the existing components
- Feb-2018 – Version 6.0, new RHEL version (RHEL 7.4), new EPICS version (EPICS 7), support for S7-1500 PLCs, CS-Studio tools adaptations for TCR and industrial systems operation.
- Mar-2019 – Version 6.2, upgrade of PostgreSQL (v10) and Eclipse (v4.7.3), further extensions for TCR deployments, programs handling in SDD, first stable version of OPC UA support.
- Oct 2020 – Version 6.3, full integration of OPC UA.
- March 2022 – Version 7.0, new RHEL version (RHEL 8.5), PostgreSQL v12, added Modbus and pyDevSup EPICS Support.
- March 2023 – Version 7.1, upgrade of CS-Studio (4.8.1) and Eclipse (v4.24).
- March 2024 – Version 7.2, upgrade of CS-Studio (4.9.1), inclusion of the Real-Time Framework (RTF) and Nominal Device Support v3 (NDSv3) components.

2 CODAC CORE SYSTEM

2.1 Overview

CODAC designates the central control system responsible for operating ITER.

The different plant systems that constitute the ITER device will each be driven by local instrumentation and control, designated as a *Plant System I&C*.

A plant system I&C contains local controllers that are either:

- standardized programmable logic controllers, designated as PLCs or *Slow Controllers*,
- standardized rack mounted computers which may control PCI/PCIe/cRIO input/outputs and are designated as PCFs (plant controller, fast) or *Fast Controllers*,
- non-standardized controllers, designated as POCs (*Plant Other Controllers*).

Each plant system I&C also includes exactly one *Plant System Host (PSH)*, supplied by the ITER Organization (IO), which implements standard functions not requiring plant system specific software nor any signal inputs/outputs.

The plant system I&C physical architecture is illustrated in Figure 2-1.

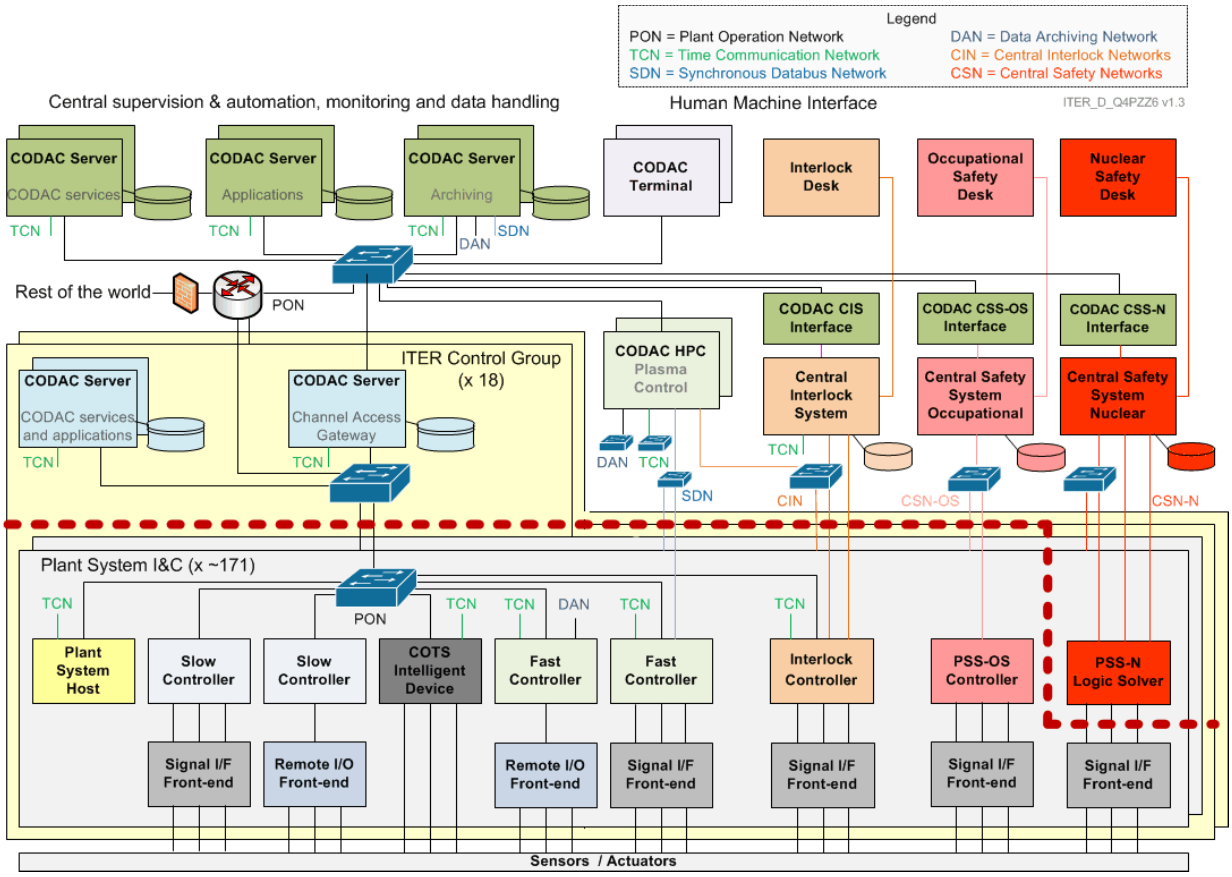


Figure 2-1 – Plant System I&C Physical Architecture

During development and testing, the CODAC central infrastructure, including servers and operator terminals, is replaced by a computer designated as *Mini-CODAC*, which implements a subset of the central CODAC functions.

CODAC Core System is a software package that is distributed by the ITER Organization for the development of the plant system I&C. It includes the software for the different types of I&C

computers and provides the plant system I&C developers with the environment required to develop and test the software in a way that complies with the ITER requirements.

CODAC Core System comes bundled together with the Red Hat Enterprise Linux (RHEL) Operating System, preconfigured for the CODAC Core System.

The CODAC Core System software layer is based on **EPICS** [RD2], an open-source distributed control system software platform that is widely used in experimental physics. The EPICS framework and utilities are developed and maintained by its user community, which ITER is a part of.

2.2 Release Scheme

All CODAC Core System releases are labelled with a sequence of three numbers (the CODAC Core System version identifier) that identify the major branch (or release series), the minor variant, starting from 0 for each major version, and the maintenance iteration, starting from 0 for each minor release. For example, CCS version 6.3.2 is the second maintenance release for the 6.3 version, while CCS version 7.2 is second minor release of version 7.

Since 2017, one version per year is planned, with a nominal release date in February. In addition, maintenance releases are issued when cumulated bug fixes need to be distributed to many users and deployed on many systems, usually in fall.

Before, on a faster development cycle, two official CODAC Core System (minor) releases were published every year: one to introduce major changes and new features, the other one improving and extending it.

Patches to fix known issues are produced on demand and distributed to affected users.

Preliminary versions (alpha and beta releases) are produced before any official version. They are only distributed to developers and testers and shall not be used for I&C development.

Two or more different minor versions can be installed simultaneously on a computer, provided their major version is the same. Only one of them can be active at any time. This feature is intended to facilitate migration to the latest minor version of a release series.

It is strongly recommended to migrate to the latest version of CODAC Core System. Tools are provided for upgrading systems and for migrating applications. The support team can also assist users with such a migration.

With each release, a list of enhancements and bug fixes is recorded. This list is published in the release notes; more details can be obtained from CODAC Support. Generally, there is no retrofitting of bug fixes to previously released versions, but patches can be provided for specific users/installations if required.

3 USERS, DISTRIBUTION, SUPPORT AND TRAINING

3.1 ITER Users

Users who contribute to the development of the ITER I&C System, e.g., ITER Domestic Agencies, research institutes associated with them or industry partners working for ITER through contracts, may become registered CODAC Core System users, or in short, *registered users*.

The procedure for becoming a registered user is described in the ITER CODAC public web pages [RD4]. Requests go through CODAC Support and are handled by the IO CODAC Responsible Officer.

Registered users have access to all official ITER software releases and to CCS User Support services.

3.2 CODAC Core System Distribution

CODAC Core System is available for distribution to all registered users through a software distribution server (Red Hat Satellite) hosted at IO premises. That server allows access to all the available software in the form of installable software packages (Red Hat RPM packages using the DNF package manager). For installation, the target needs to be registered with the distribution server. The installation procedure is fully automated and described in the CCS Installation Manual and installation website, available to all registered users.

Different profiles or roles of CODAC Core System allow configuring different target systems according to their use: mini-CODAC, PSH, PCF or CODAC terminal. These profiles are each available in two variants, one for development that includes the development tools and one for operation that is stripped down to the components required for operation.

The documentation (listed in chapter 8) is included in the software distribution, together with supporting example code.

3.3 CODAC Support

A support service is available for all registered users. It can be reached by email and operates continuously during IO opening hours.

Requests are sent to codac-support@iter.org. A dedicated team addresses them, recording requests and actions in an issue tracking system, which is monitored by the different IO ROs who contribute to the support activities and validate the actions.

Since the release of version 7.0, full support for versions 6.x is maintained, but the support for versions 5.x will be limited to migration assistance. No regular support is provided for versions 4.x or older.

3.4 Training

Hands-On workshops for ITER I&C developers (4 days) are organized at IO Headquarters on a regular basis, and occasionally at Domestic Agency premises, according to needs.

On-line training lessons (presentations and demonstrations) are also available for ITER users through the ITER On-line training centre site [RD5].

4 System Components

4.1 RHEL

Red Hat Enterprise Linux (RHEL) is the standard operating system for CODAC Core System, used on mini-CODAC, PSHs, Fast Controllers, CODAC Terminals and CODAC Servers. All of these systems use a version of RHEL that is officially supported by Red Hat.

The operating system is included in the software distribution, so that any CCS installation will first install the RHEL operating system on the target computer.

The only supported architecture is linux-x86_64, corresponding to 64-bit Intel x86 architecture. Other architectures (including 32-bit Intel) are not supported.

The following table shows which RHEL version is used for each major version of CCS.

CCS major version	RHEL version
7.x	8.5
6.x	7.4
5.x	6.3
4.x	6.1
3.x	5.5
2.x	5.3

For real time dependent applications on Fast Controllers, the MRG-Realtime (MRG-R) kernel extension is provided on an as-needed basis.

4.2 EPICS

EPICS Base is included in the CCS distribution and required for all system profiles. The EPICS toolkit is the base for control processes running on Fast Controllers and PSHs. The EPICS communication protocols, Channel Access (CA) and pvAccess (PVA), are the standard communication protocols to access plant system I&C data over the Plant Operation Network (PON).

Up until CCS version 4.x, the included version of EPICS Base was 3.14. In CCS version 5.x, the version of EPICS Base was 3.15.

In CCS version 6.x and 7.x, EPICS 7 is used. EPICS 7 extends EPICS Base version 3.x with the so-called PVA modules that include the pvData and pvAccess libraries to handle structured data over the network, using the pvAccess network protocol.

The following major EPICS components are included:

- Sequencer, for the execution of state machine programs written using State Notation Language (SNL) in PSHs, Fast Controllers and mini-CODAC systems.
- IOC log, for error and trace logging.
- Autosave, for automatically saving and reloading the values of configuration variables to and from the disk where persistence is required.
- CA and PVA Gateways, for allowing Channel Access and pvAccess clients to access servers on different sub-networks.

- Multi-Core Utilities, for configuring how EPICS threads are handled by the different CPU cores, isolating them, and tweaking priorities.
- Java implementations of Channel Access and pvAccess, for Java applications that are CODAC clients.
- Python wrappers for Channel Access and pvAccess, for Python applications that are CODAC clients.
- IOC Monitor / System Monitor, for monitoring IOC and system (host) health parameters as PVs.
- The Visual Database Configuration Tool, for configuring EPICS databases using a GUI.
- ASYN, a generic driver framework for developing EPICS Device Support.
- Stream Device, a generic Device Support module that allows writing stream based data exchange protocols between EPICS records and hardware.
- Area Detector, a generic Device Support module used for image acquisition devices.
- OPC UA Device Support, a Device Support module allowing IOCs to interface OPC UA servers.
- Modbus Device Support to allow interfacing devices through the Modbus protocol.
- pyDevSup, a generic Device support module that allows to integrate devices using Python scripts.

In addition, Device Support modules for all supported hardware from the Fast and Slow Controller catalogues are provided.

4.3 Other System Components

The following software components are also included and configured in the distribution:

- A **PostgreSQL** relational database, containing the following databases:
  - SDD, for storing plant system I&C data
  - Alarm, for storing alarm server configuration and alarm history
  - Archive, for storing archive engine configuration and archived data
  - Logbook, for storing events.
- The **Eclipse** platform used for specific editors, SDD Editor and Maven Editor, as well as for the Control System Studio tools. Plug-ins for development with C/C++, Java, Python, and Subversion are also included.
- **Apache ActiveMQ** as a JMS implementation for Control System Studio.

5 Development tools

5.1 EPICS Records

EPICS includes a framework for developing control processes, called IOCs (Input Output Controllers). Each IOC process is built around a database of records that belong to a set of pre-defined record types.

The EPICS build tools have not been modified, but are encapsulated inside ITER specific commands (cf. ITER Maven Tools, below). The development of EPICS configuration files should be done using the SDD tools, in order to satisfy the requirements for integration. This includes enforcing of naming conventions and conventions on files that imported EPICS software shall satisfy.

5.2 Self-Description Data

5.2.1 The Self-Description Data Concept

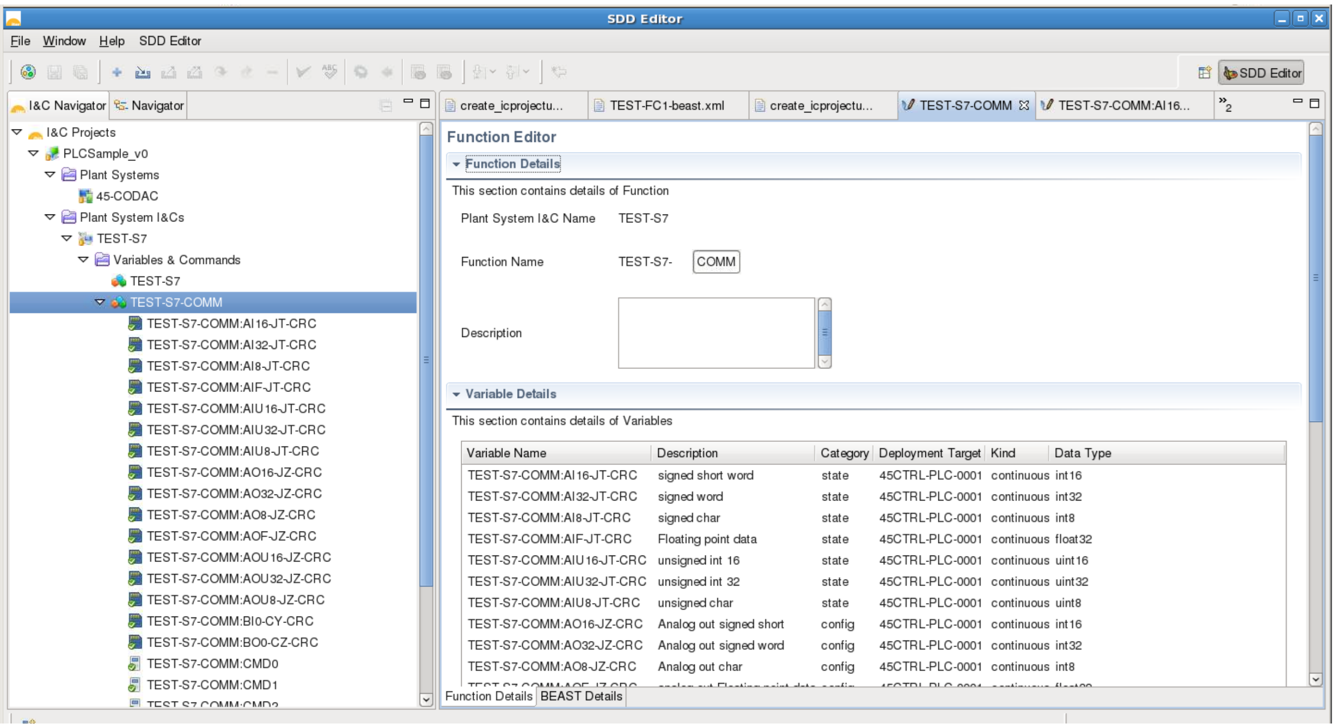
Self-Description Data (SDD) is an ITER concept covering the static data that configures the plant system I&C. This data is produced by the plant system I&C designers and developers, using IO tools and according to an IO defined schema.

The data will be delivered by the plant system I&C developers into an IO database upon software delivery. During development, it is stored in a local database, the SDD database.

The SDD Toolkit has been developed by IO to allow the user to configure the plant system I&C. It includes:

- The **SDD Editor** to define the plant system interface, the I&C components, the interfaced signals and to configure variables, alarms, archiving, etc.

  The SDD editor is an Eclipse RCP application integrated with the other development tools and operating on a local SDD database .
- The **SDD Translator** to convert the SDD data into the required configuration data for mini-CODAC, PSH and Fast Controllers and to generate the required STEP 7 files for interfacing PLCs.
- The **SDD Sync** tool to save and load SDD data to/from XML files.
- The **SDD Parser** to parse user-provided or user-modified configuration files, e.g., EPICS database definitions, and retrofit the changes into the SDD database.



- I&C Projects
  - PLCSample_v0
    - Plant Systems
      - 45-CODAC
    - Plant System I&Cs
      - TEST-S7
        - Variables & Commands
          - TEST-S7
          - TEST-S7-COMM
            - TEST-S7-COMM:AI16-JT-CRC
            - TEST-S7-COMM:AI32-JT-CRC
            - TEST-S7-COMM:AI8-JT-CRC
            - TEST-S7-COMM:AIF-JT-CRC
            - TEST-S7-COMM:AIU16-JT-CRC
            - TEST-S7-COMM:AIU32-JT-CRC
            - TEST-S7-COMM:AIU8-JT-CRC
            - TEST-S7-COMM:AO16-JZ-CRC
            - TEST-S7-COMM:AO32-JZ-CRC
            - TEST-S7-COMM:AO8-JZ-CRC
            - TEST-S7-COMM:AOF-JZ-CRC
            - TEST-S7-COMM:AOU16-JZ-CRC
            - TEST-S7-COMM:AOU32-JZ-CRC
            - TEST-S7-COMM:AOU8-JZ-CRC
            - TEST-S7-COMM:BI0-CY-CRC
            - TEST-S7-COMM:BO0-CZ-CRC
            - TEST-S7-COMM:CMD0
            - TEST-S7-COMM:CMD1

Function Editor

Function Details

This section contains details of Function

Plant System I&C Name TEST-S7

Function Name TEST-S7- COMM

Description

Variable Details

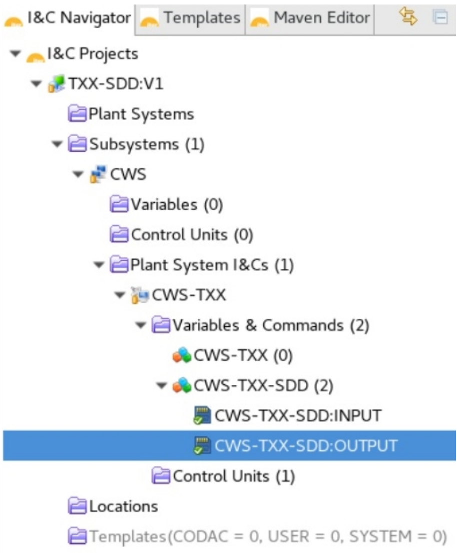
This section contains details of Variables

Variable Name	Description	Category	Deployment Target	Kind	Data Type
TEST-S7-COMM:AI16-JT-CRC	signed short word	state	45CTRL-PLC-0001	continuous	int16
TEST-S7-COMM:AI32-JT-CRC	signed word	state	45CTRL-PLC-0001	continuous	int32
TEST-S7-COMM:AI8-JT-CRC	signed char	state	45CTRL-PLC-0001	continuous	int8
TEST-S7-COMM:AIF-JT-CRC	Floating point data	state	45CTRL-PLC-0001	continuous	float32
TEST-S7-COMM:AIU16-JT-CRC	unsigned int 16	state	45CTRL-PLC-0001	continuous	uint16
TEST-S7-COMM:AIU32-JT-CRC	unsigned int 32	state	45CTRL-PLC-0001	continuous	uint32
TEST-S7-COMM:AIU8-JT-CRC	unsigned char	state	45CTRL-PLC-0001	continuous	uint8
TEST-S7-COMM:AO16-JZ-CRC	Analog out signed short	config	45CTRL-PLC-0001	continuous	int16
TEST-S7-COMM:AO32-JZ-CRC	Analog out signed word	config	45CTRL-PLC-0001	continuous	int32
TEST-S7-COMM:AO8-JZ-CRC	Analog out char	config	45CTRL-PLC-0001	continuous	int8

Figure 5-1 – SDD Editor: List of Variables in a Function of a Plant System I&C

5.2.2 SDD Content

Information about the following elements can be included in the SDD of an I&C project:



- I&C Projects
  - TXX-SDD:V1
    - Plant Systems
    - Subsystems (1)
      - CWS
        - Variables (0)
        - Control Units (0)
        - Plant System I&Cs (1)
          - CWS-TXX
            - Variables & Commands (2)
              - CWS-TXX (0)
              - CWS-TXX-SDD (2)
                - CWS-TXX-SDD:INPUT
                - CWS-TXX-SDD:OUTPUT
            - Control Units (1)
    - Locations
    - Templates(CODAC = 0, USER = 0, SYSTEM = 0)

- Plant Systems (PBS…)
- Cubicles (location…)
- Chassis (location…)
- Control units (type, network interfaces…)
- I/O modules (type…)
- Components (type…)
- Signals associated with components (direction, type, logic, range…)
- Control groups (CBS…)
- CODAC Servers (network interfaces…)
- Plant System I&Cs (CBS…)
- Plant System Operating State machine (PSOS) states and transitions, and how they are mapped to those of the Common Operating State machine (COS)
- I&C programs (deployment target, variables interfaced, I/O modules controlled…)
- I&C functions (CBS…)
- Variables, including:
  - Tier (Conventional, Interlock, Safety)
  - Network (PON, SDN, DAN)
  - Signal mapping
  - EPICS/SDN/DAN attributes (for PON/SDN/DAN variables, respectively)
  - Alarm attributes

- Archiving attributes
- Pulse scheduling attributes

5.3 The Development Workflow

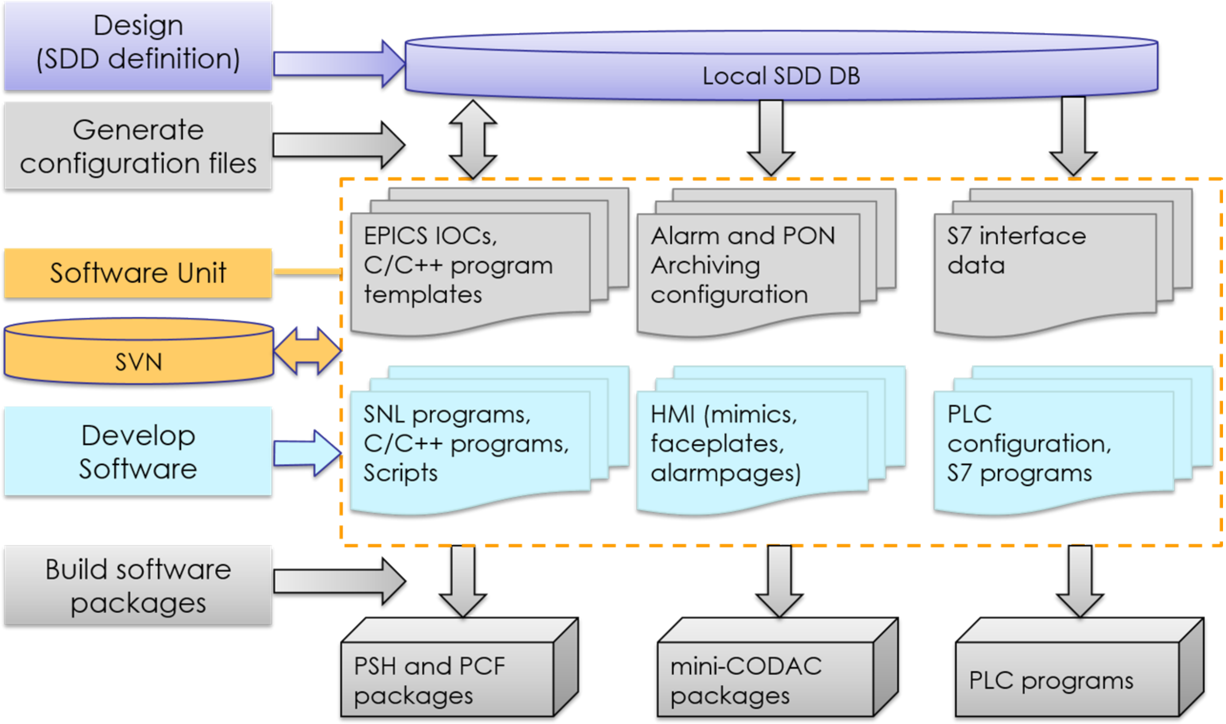

Design
(SDD definition)
Local SDD DB
Generate configuration files
Software Unit
EPICS IOCs, C/C++ program templates
Alarm and PON Archiving configuration
S7 interface data
SVN
Develop Software
SNL programs, C/C++ programs, Scripts
HMI (mimics, faceplates, alarmpages)
PLC configuration, S7 programs
Build software packages
PSH and PCF packages
mini-CODAC packages
PLC programs

Figure 5-5-2 – The Development Workflow

The development workflow, as illustrated in Figure 5-5-2, consists of the following steps:

1) Define all configuration with SDD tools, including:
   – the variables with their EPICS details for building the IOC process
   – the programs
   – the functions (CBS nodes for variables, OPIs, alarms, etc.)
   – the configuration for alarms, archiving.
2) Generate the configuration files.
3) Create the initial version of the software unit.
4) Develop any additional software for the project as required, such as:
   – the code for the C/C++ programs
   – state machines in SNL or other scripts
   – operator Interfaces
   – the code for the PLCs.
5) Build the software unit, creating the binary RPM packages to be deployed on the different target computers.

5.4 ITER Maven Tools

The development workflow is supported by the SDD tools and by commands implemented using the Apache Maven build tool.

These commands implement all the steps for building and testing the software unit. They also allow saving, restoring and synchronizing the project files with a Subversion (SVN) source repository (cf. next section).

A graphical tool, the Maven Editor, provides the user with a graphical HMI for executing these commands – either as a stand-alone tool or by means of a dedicated view in the SDD Editor.

As illustrated in Figure 5-3, the Maven Editor interfaces allow the user to build the software, run the executables for test purposes on local or remote hosts and build the RPM packages for deployment.

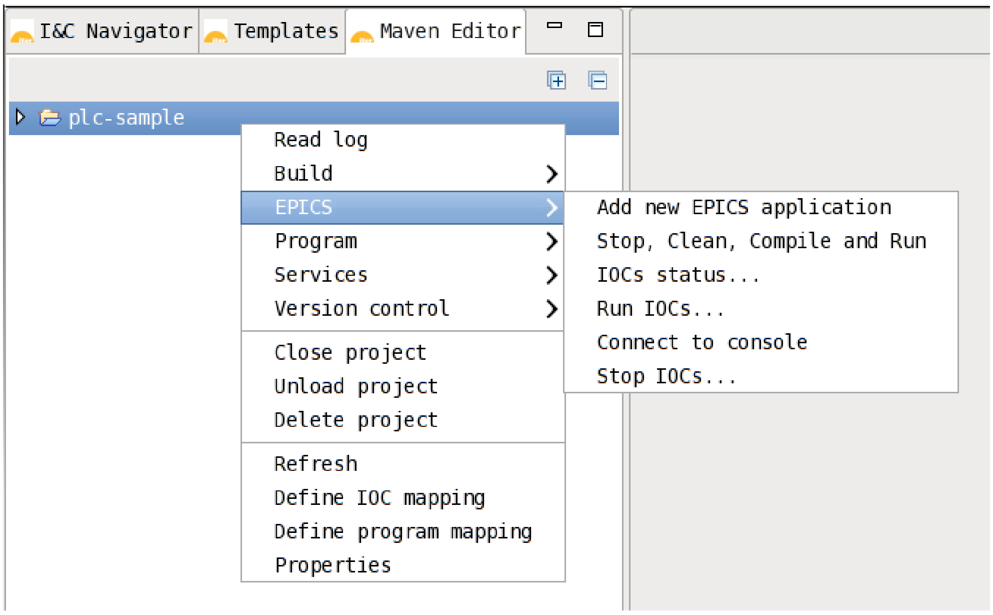


Figure 5-3 – Maven Editor

5.5 Source Code Repository

ITER provides a Subversion (SVN) server as a central service to all ITER I&C developers for maintaining any software product for I&C. The nodes in the central repository are created by IO for receiving the software associated with a PA or contract. The suppliers are in charge of delivering and tagging the I&C software.

The I&C software shall be delivered in source format for allowing it to be re-built using appropriate procedures before deployment.

The main commands for exchanging source files with the central repository are:

- **Import** – takes a local directory that is not synchronized with any SVN location, and performs an initial import to a specified SVN location..
- **Checkout** – the reverse of import. Takes a directory on the central SVN repository and makes a copy on the local system.
- **Update** – checks the central SVN repository and, if changes have been made, it updates the local copy with those changes.
- **Commit** – performs the reverse operation as update. Changes made on the local copy are pushed as a new revision to the central SVN repository.

All these commands are integrated into Maven Editor, so that they can be easily executed with any I&C software unit.

6 CODAC SERVICES

6.1 Mini-CODAC

During plant system I&C development, the mini-CODAC system has two roles:

- A local CODAC Server implementing the following “central” CODAC services:
  - operator interface,
  - alarm handling,
  - data archiving,
  - electronic logbook,
  - error & trace logging
- A development workstation with the SDD database and other shared files or file systems.

Development and testing, including Factory Acceptance Tests (FAT), shall be executed using the mini-CODAC as a replacement for CODAC.

After integration, CODAC services will be provided from CODAC servers and the development tools shall be run on those systems for the maintenance tasks. The configuration data produced with mini-CODAC will be re-used in this final configuration.

6.2 Control System Studio

Control System Studio (CS-Studio) is an Eclipse-based suite of tools developed within the EPICS community. It provides an integrated environment based on Eclipse plug-in technology and is therefore easily extensible.

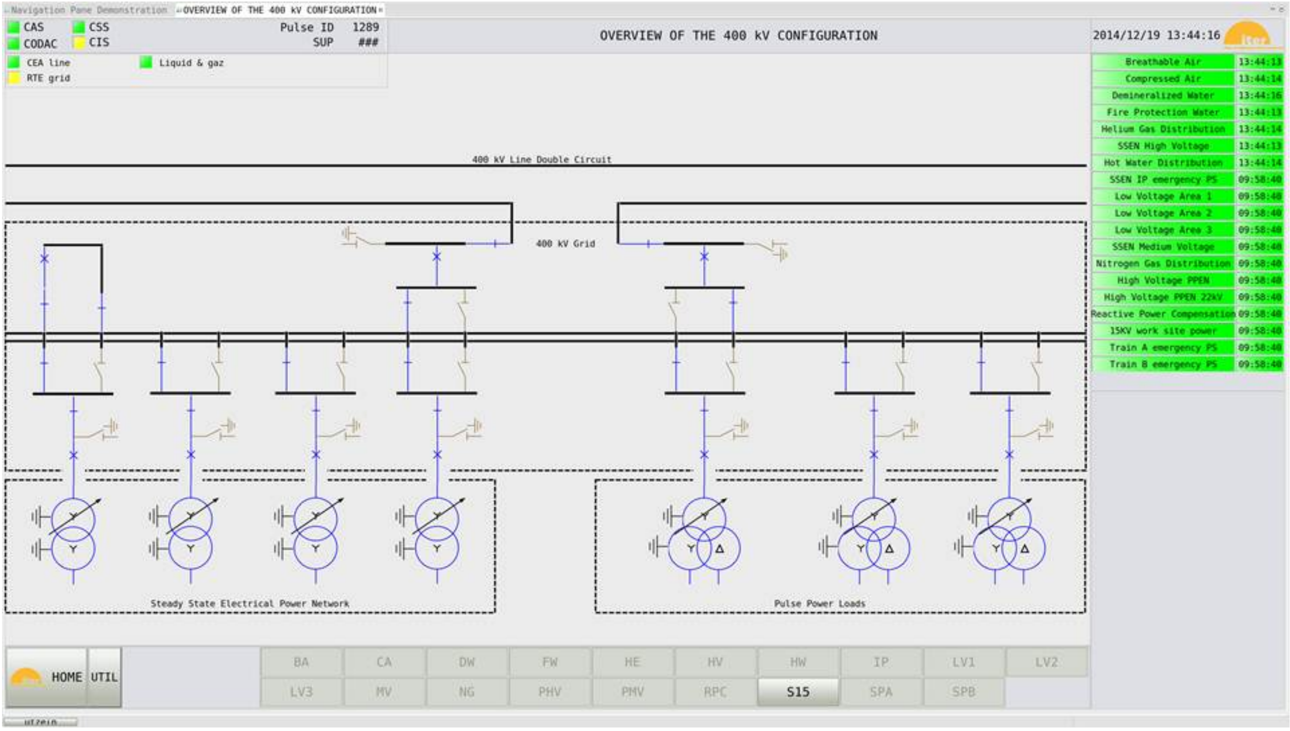
Tools developed in this environment have been adopted for the mini-CODAC services: HMIs, alarms and archiving.

6.3 Operator Interface Development

The display manager **BOY**, developed with the CS-Studio platform, allows the development of dynamic displays using an interactive editor and the run-time management of such displays. In addition to its integration into the CS-Studio environment, BOY offers a rich set of graphical widgets and dynamic configuration features that can be both extended and customized to satisfy specific requirements.

A dedicated library of industrial symbols, with graphical widgets for display and dynamic attributes, has been added for ITER applications (I&C OPIs).

BOY supports the standardized ITER HMI Design Guide for OPI templates, fonts and colours, as illustrated in Figure 6-1.


Navigation Pane Demonstration
OVERVIEW OF THE 400 kV CONFIGURATION
🟩 CAS
🟩 CSS
🟩 CODAC
🟨 CIS
🟩 CEA line
🟩 Liquid & gaz
🟨 RTE grid
OVERVIEW OF THE 400 kV CONFIGURATION
400 kV Line Double Circuit
400 kV Grid
Steady State Electrical Power Network
Pulse Power Loads
HOME
UTIL
BA
CA
DW
FW
HE
HV
HW
IP
LV1
LV2
LV3
MV
NG
PHV
PMV
RPC
S15
SPA
SPB

Pulse ID 1289
SUP ###

2014/12/19 13:44:16

Breathable Air	13:44:13
Compressed Air	13:44:14
Demineralized Water	13:44:16
Fire Protection Water	13:44:13
Helium Gas Distribution	13:44:14
SSEN High Voltage	13:44:13
Hot Water Distribution	13:44:14
SSEN IP emergency PS	09:58:40
Low Voltage Area 1	09:58:40
Low Voltage Area 2	09:58:40
Low Voltage Area 3	09:58:40
SSEN Medium Voltage	09:58:40
Nitrogen Gas Distribution	09:58:40
High Voltage PPEN	09:58:40
High Voltage PPEN 22kV	09:58:40
Reactive Power Compensation	09:58:40
15KV work site power	09:58:40
Train A emergency PS	09:58:40
Train B emergency PS	09:58:40

Figure 6-1 – OPI Compliant with ITER HMI

CS-Studio is not only used for developing OPIs, but also for running them. Running an OPI means having its elements or widgets animated in real-time with the process data and allowing the operators to interact with the plant system.

OPIs developed for the plant system I&C shall be available for monitoring and controlling the equipment. In addition, some simple tabular PV OPIs are generated for all CBS functions that include all variables of the function. These OPIs can also be used for application testing and verification.

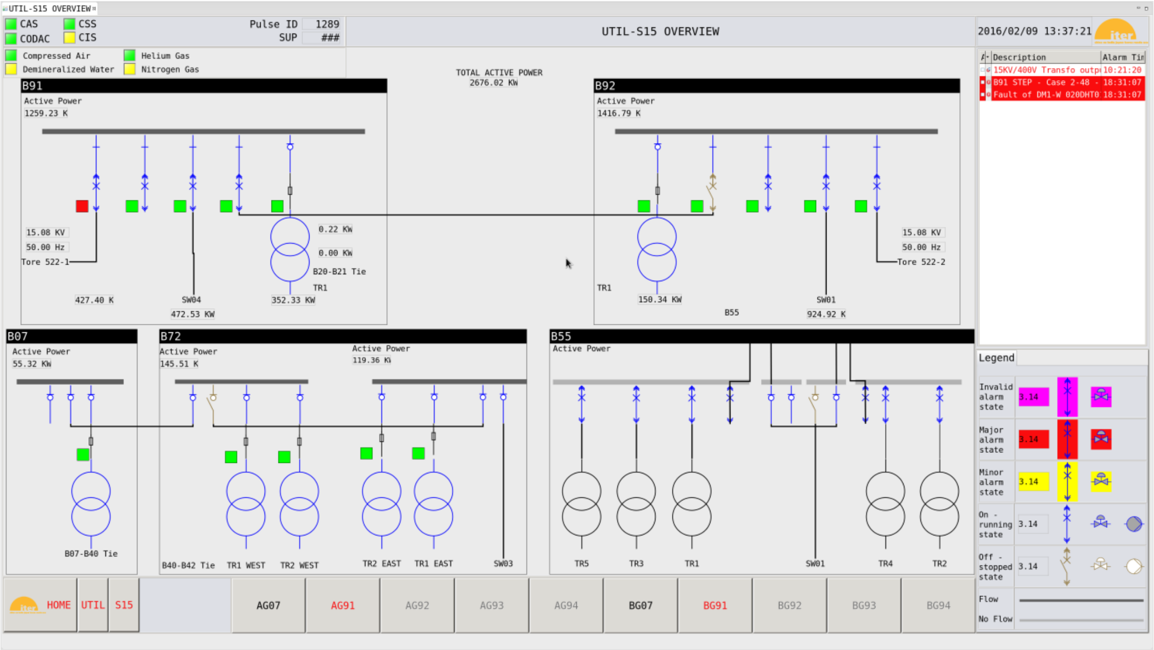
CODAC HMIs consist of a display for each CBS function, a navigation pane to navigate between the CBS functions, an alarm pane displaying any triggered alarms, component faceplates and alarm pages for every alarm enabled variable.

6.4 Alarm Handling

BEAST is a distributed alarm system consisting of alarm servers that monitor alarms from the EPICS IOC processes via Channel Access and a user interface for viewing the current alarms as well as acknowledging them and browsing the history of alarm messages.

An alarm can also trigger the execution of automated actions, including sending an email or submitting a new logbook entry.

Alarm logs and configurations are stored in the included relational database.



A		Description	Alarm Tim
		15KV/400V Transfo outp	10:21:20
		B91 STEP - Case 2-48 -	18:31:07
		Fault of DM1-W 020DHT0	18:31:07

Figure 6-2 – Alarm Handler (BEAST) User Interface

Alarm displays are fully integrated into the CODAC HMI. They are visible to operators in a variety of ways:

- The top right window of the HMI contains the Alarm Pane, where all the triggered alarms, current and unacknowledged are listed. Right-clicking on an alarm in the alarm pane provides quick access for operators to execute several actions, such as:
  - Acknowledging the alarm
  - Opening an alarm page to get detailed information about the alarm
  - Triggering a pre-configured command
  - Opening windows with pre-configured guidance messages.
- Widgets in the mimic linked to a PV may have a current alarm change colour to reflect their alarm severity.
- Nodes on the navigation pane change colour to display the highest unacknowledged alarm severity at their CBS level or below.

Alarm descriptions can also be automatically annunciated by a text-to-speech converter.

6.5 Data Archiving

The EPICS data archiver **BEAUTY** is included in the CCS distribution. It allows deploying archive engines that save process value data into a relational database.

A data viewer, the **CS-Studio DataBrowser,** allows retrieving and viewing archived data as well as monitoring of live data. Archived data trends can also be viewed in OPIs using the Archive XY Graph widget.

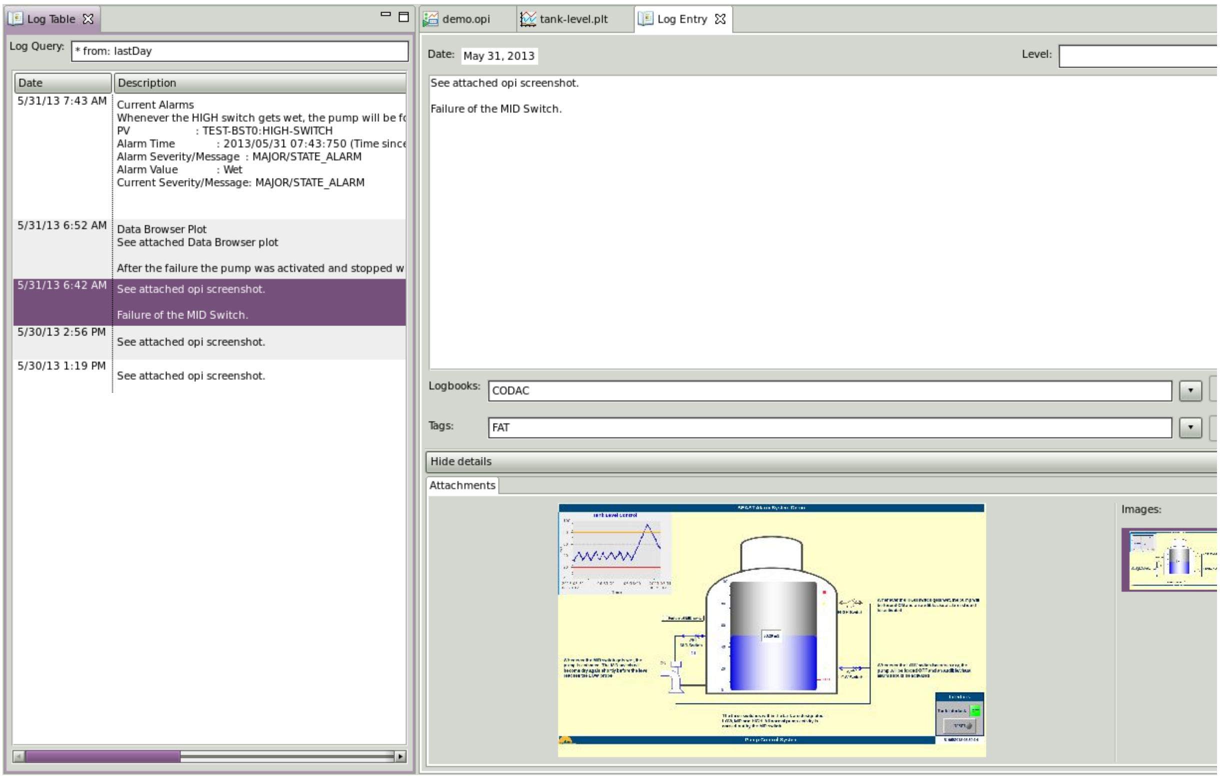
Data archiving requires the set of variables to archive to be configured with SDD tools.

Archived data is also pushed to data servers to be accessible by means of the Unified Data Access (UDA) retrieval layer.

6.6 Electronic Logbook

An electronic **Logbook** is available in the operational configuration of CS-Studio and integrated with the other CS-Studio tools.

Formatted text and screenshots can be included in logbook entries. OPIs, alarm views and data plots can be inserted on demand. Logbook entries are also created from alarms in an automatic manner.



Log Query: * from: lastDay

Date	Description
5/31/13 7:43 AM	Current Alarms Whenever the HIGH switch gets wet, the pump will be fc PV : TEST-BST0:HIGH-SWITCH Alarm Time : 2013/05/31 07:43:750 (Time since Alarm Severity/Message : MAJOR/STATE_ALARM Alarm Value : Wet Current Severity/Message: MAJOR/STATE_ALARM
5/31/13 6:52 AM	Data Browser Plot See attached Data Browser plot After the failure the pump was activated and stopped w
5/31/13 6:42 AM	See attached opi screenshot. Failure of the MID Switch.
5/30/13 2:56 PM	See attached opi screenshot.
5/30/13 1:19 PM	See attached opi screenshot.

Date: May 31, 2013

Level:

See attached opi screenshot.

Failure of the MID Switch.

Logbooks: CODAC

Tags: FAT

Figure 6-3 – Log Entry

7 CONTROLLERS SUPPORT

7.1 System Monitoring

System monitoring features are included for monitoring the status of the controllers as well as the status of the control tasks (EPICS IOC processes) running in these control units.

Each IOC built using CCS includes a monitoring module, providing, by means of Process Variables, information about the IOC process (process ID, start time, port used…) and its resource usage (CPU, memory, internal queue usage, number of remote clients…).

In addition, for each PSH and controller, a dedicated IOC process is built to provide information about the host computer (time accuracy, boot time, software version) and its resources (CPU load, memory usage, file descriptors, status of I/O boards). A set of Process Variables provides the user with an interface to this data.

Diagnostic displays using these variables can be developed or generated in an automated manner. The system monitoring variables are also intended to be used for generating alarms when errors are detected on processes or computers.

The monitoring of the cubicle environment (e.g., temperature, door status…), implemented by dedicated PLCs in standard cubicles, is also interfaced through system monitoring variables.

7.2 Support for PLCs

CODAC Slow Controllers are standardized as Siemens S7 PLCs. The recommended components are described in the ITER catalogue for I&C products – Slow controllers PLC [RD7]. Siemens S7 PLCs do not run CODAC Core System, as they use their own system environment. The tool used to program these PLCs should be either Siemens Simatic Manager or TIA Portal. These tools are not included in the CODAC Core System distribution.

However, the CODAC Core System distribution does include the tools to build the EPICS software that is necessary for integrating the PLCs into the CODAC infrastructure. Communication with the PLCs is implemented by EPICS IOC processes deployed on the PSH or alternatively on other controllers running CCS.

The communication is synchronizing EPICS variables on the PSH with the PLC variables. The PLC variables must be defined using the SDD tools. Each PLC command or variable is mapped to an EPICS process variable in the PSH and to a program variable in the PLC.

CODAC PLCs must synchronize their system time using NTP and use the PSH as a time source.

7.2.1 Integration using SPSS

When using this option, PLC S7 programs must be built based on a predefined program provided with the CCS distribution called Standard PLC Software Structure (SPSS). The SPSS essentially establishes the communication channels with a PLC IOC and provides a program structure on which the final program should be built. The SDD tools generate the variables exchanged with the EPICS IOC in dedicated data blocks that are handled by the SPSS-based programs.

Automatically generated data are also inserted in the transmitted frames (header, trailer, length, application version number, timestamp, heartbeat) in order to detect communication errors and data inconsistencies. This data is also generated by the SDD tools.

Redundant PLCs (used for example in interlock systems) are supported. This support includes the transparent connection to the active master CPU and the reconnection to the new master in the case that a switch-over is triggered because of a failure.

7.2.2 Integration using OPC UA

When using this option, an OPC UA server must be running on top of the Slow Controller. On the S7-1500 series, this can be enabled in the TIA Portal software after a specific OPC UA server software license has been acquired for the PLC.

Under OPC UA, variables on the PLC are accessible as OPC UA *nodes* through a unique *node ID*, a combination of a string identifier and a namespace number. The OPC UA representation of PLC variables is visible and configurable in TIA Portal; the node IDs have to be entered in SDD to correctly create the EPICS IOC configuration representing the PLC variables.

From the Siemens S7 PLCs in the ITER Slow Controller catalogue, only the models of the 1500 series are providing an embedded OPC UA server. Redundant PLCs (used for example in interlock systems) are currently not supported.

7.3 Timing and Synchronization

IEEE-1588, also called the Precision Time Protocol (PTP), has been selected for the Time Communication Network (TCN).

PXI boards as well as Ethernet interfaces compliant with the PTP protocol are supported for Fast Controllers.

The timing and synchronization board for Fast Controllers is **NI PXI-6683H**.

The software support includes Linux drivers, EPICS drivers, SDD integration, test programs and documentation.

The following features are supported:

- Accurate system time using PTP. The accuracy depends on the local infrastructure. With a proper master clock and distribution, it will be better than 1 µs.
- Generation of events and clock signals using the precise time distributed by the TCN.
- Time-stamping incoming signals.

On the ITER TCN network, the accuracy of the board time will be 50 ns RMS.

The TCN library provides C/C++ developers with functions for waiting for an ITER time or registering actions synchronized with ITER time.

7.4 Support for I/O Boards

The hardware that is recommended by IO for Fast Controllers is published in the ITER Catalog of I&C products – Fast Controllers [RD8]. This catalog is a satellite document of the PCDH [RD1]. The scope of I/O board software support is limited to the boards that are defined in this catalog and are adopted for plant system I&C. The software support includes Linux drivers, EPICS drivers, SDD integration, test programs and documentation.

All National Instruments modules referenced in this document are supported.

These are:

- The **NI PXIe-6368** board (X-series), which is a high performance multi-function data acquisition board.
  - 16 differential Analog Input channels (16 bits), 2 MSamples/s
  - 4 Analog Output channels (16 bits), 3.3MS/s
  - 48 bi-directional Digital channels
  - 4 general purpose counters (32 bits).

  The **NI PXIe-6363** board, a lower-cost version with reduced performance (no simultaneous sampling) is also supported, starting with CCS 6.0.
- A digital I/O board, **NI PXI-6528**, with the following channels:
  - 24 optically isolated input channels
  - 24 solid-state relay output channels.
- The **NI FlexRIO** PXIe-796xR bundles, composed of an FPGA board with an adapter module for analog I/O, digital I/O or camera interface.
- An older deprecated multi-function data acquisition board, **NI PXI-6259**, with the following channels:
  - 16 differential / 32 single-ended Analog Input channels (16 bits), 1 MSamples/s total for multi-channel, 1.25 MSamples/s for a single channel
  - 4 Analog Output channels (16 bits), 1.25MS/s for 4 channels, 2.86 MS/s for 1 channel.
  - 48 bi-directional Digital channels
  - 2 general purpose counters (32 bits).

  New developments should consider using the NI-PXIe-6363 (mentioned above).
- The **NI CompactRIO** modules described in the Fast Controllers catalog.

7.5 SDN Support

The Synchronous Databus Network (SDN) is the high performance real time network to be used for sharing of real time data across ITER plant systems and with the Plasma Control System (PCS).

The SDN software provides communication services following an anonymous publish-subscribe design pattern. Participants exchange data over logical communication channels, i.e., topics, without the necessity of being aware of other participants.

The data exchanged via SDN is defined using SDD tools as SDN topics. The programs deployed on Fast Controllers can be configured for publishing data on SDN topics or for subscribing to them.

The SDN Application Programming Interface (API) is included in the CCS distribution. It comprises a library for C++ programs that implements all the required services for SDN communications.

An SDN monitoring service can also be started for gathering information on SDN data exchanges.

7.6 DAN Support

The Data Archiving Network (DAN) is the dedicated network for streaming high volume data, such as scientific data derived from diagnostic systems. Data transmitted by the distributed producers are received by a DAN archiver and saved into HDF5 files. A data access library and a set of plotters allow retrieving and visualization the saved data. The data can also be retrieved by means of the Unified Data Access (UDA) layer.

DAN data is defined using SDD tools and processes deployed on Fast Controllers can be configured for publishing on DAN. The distribution includes the DAN Application Programming Interface (API) that allows processes to do that.

A DAN streamer process is provided to be deployed on each Fast Controller for collecting data from the different processes and pushing it over DAN.

7.7 Common Operating State Support

COS variables are a mandatory part of the Plant System I&C interface with CODAC. They provide the supervision (SUP) system with a standard interface for each Plant System I&C. The variables allow acquiring the current state and requesting a transition.

The COS variables are automatically added to any Plant System I&C. Their implementation is configured by means of a mapping table between the generic COS variables and the Plant System specific variables that the I&C developer should implement. These specific variables can be implemented either on a Fast Controller or on a PLC.

7.8 Real-Time Framework (RTF)

The ITER Real-Time Framework (RTF) is a software framework for building real-time applications such as the Plasma Control System (PCS) and plasma diagnostics. It is entirely written in C++ and it runs real-time control loops on isolated CPU threads, adapting busy-wait synchronization techniques to optimize response time and jitter.

The deployment of RTF applications can be local or distributed and is configured separately by assigning blocks to threads, processes and nodes. Data is exchanged synchronously using Gateway function blocks (implicitly) inserted on both communication ends. For inter-thread communication, these gateways are implemented with dedicated real-time queues, while for the inter-process and inter-node communication an RTF Transport Layer implementation can be used (SDN, EPICS, DAN, NDS, etc).

7.9 Nominal Device Support (NDSv3)

To support software integration of diverse data acquisition and timing hardware devices in diagnostics applications, the Nominal Device Support (NDSv3) C++ framework, conceived by Cosylab and under development at ITER, uses a layered approach, abstracting specific hardware device APIs as well as the interface to control systems and real-time applications.

CODAC developed NDS device drivers using both PCI express extension for instrumentation (PXIe) and Micro Telecommunications Computing Architecture (MTCA) platforms for multi-function data acquisition (DAQ) devices, timing cards and field-programmable gate array (FPGA) based solutions.

These drivers can be re-used and extended to create targeted diagnostics applications based on the mentioned platforms.

7.10 Support for C/C++ Programs

In addition, I&C functions can be implemented by means of C/C++ programs.

These are configured using SDD tools and developed using ad-hoc templates. For each I/O board, a C API is available for direct usage from programs.

C/C++ APIs are provided for

- precise time synchronization and stamping using TCN,
- synchronous data exchange over SDN
- scientific and diagnostic data production over DAN
- error and trace logging.

The interface with CODAC is implemented by means of variables declared with SDD that are generated as EPICS PVs with interface code for the programs.

8 DOCUMENTATION

The following documents are included in the CCS distribution. These are also available from the ITER documentation management system (IDM). Versions are given in the release notes.

Overview

ITER_D_34SDZ5	CODAC Core System Overview **(This document)**
ITER_D_43PSH9	CODAC Core System User Manual

Installation & Support

ITER_D_33JNKW	CODAC Core System Installation Manual
ITER_D_7JCFUD	CODAC Core System Migration Guide

Development & Testing

ITER_D_33T8LW	CODAC Core System Application Development Manual
ITER_D_32Z4W2	SDD Editor User Manual
ITER_D_46AAXR	SDD Synchronization Guide
ITER_D_A4WQDZ	How to include a new I/O module in SDD
ITER_D_7MT2YC	Maven Editor User Guide
ITER_D_35XFCY	System Health Monitoring Variables
ITER_D_QEK784	Logging library - Software User Manual

Control System Studio

ITER_D_QVBYD8	CODAC Core System CS-Studio User Guide
ITER_D_3XLESZ	HMI Style Guide and Toolkit
ITER_D_7367JQ	Operator Interface standardisation - CSS BOY Edition and Runtime
ITER_D_A69URK	Operator Interface standardisation - CSS BOY Industrial Symbol Library

Slow Controllers (PLC) and OPC UA

ITER_D_3QPL4H	PLC Software Engineering Handbook
ITER_D_G4UMX5	Standard PLC Software Structure (SPSS) User Manual
ITER_D_PJAHXJ	s7PLCAsyn EPICS Driver User's Manual
ITER_D_2N8C3M	PLC Sample Application Guide
ITER_D_32N6MJ	OPC UA Integration User Manual

Fast Controllers – IO

ITER_D_2PLQ4P	NI Sync Linux Device Driver User's Guide
ITER_D_33Q5TX	NI Sync EPICS Driver User's Guide
ITER_D_32GTJY	NI PXI-6259 Linux Driver User's Guide
ITER_D_3DEY52	NI PXI-6259 EPICS Driver User's Guide
ITER_D_3ZHXQ9	NI PXI-6528 Linux Device Driver User's Guide
ITER_D_433VEW	NI PXI-6528 EPICS Driver User's Guide
ITER_D_3LTMR6	NI X-Series Linux Device Driver User's Guide
ITER_D_3P4N3R	NI X-Series EPICS Driver User's Guide
ITER_D_LW3UFH	NI-RIO Device Driver User Manual
ITER_D_RAJ9P8	NI-RIO EPICS Device Driver User Manual
ITER_D_RATM8Z	IRIO Library User's Manual
ITER_D_QQMYTY	IRIO Design Rules for LabVIEW for FPGA
ITER_D_7ZQ27R	CODAC PXI Config Users Guide
ITER_D_S37HD2	NDSv3 Core Library User Manual
ITER_D_S977XP	NDSv3 EPICS Device Support User Manual

Fast Controllers – HPN

ITER_D_N4XTGG	TCN API – Software User Manual

ITER_D_U2TTSZ	PTPd User Manual
ITER_D_MUYNT6	TCNd User Manual
ITER_D_B7SKFU	SDN Software User Manual
ITER_D_Q6GULS	DAN User Manual
ITER_D_SD29MG	SDN Archiver User Manual

Operational Applications

ITER_D_RUZCJ9	User Manual for CODAC Operational Applications
ITER_D_RGSWHS	SDN Tools User Manual
ITER_D_S94S4T	SDN Monitor User Manual
ITER_D_2N2ZH8	RTF User Manual
ITER_D_2Q7QFZ	RTF Development Manual
ITER_D_TPLTKG	UDA User Manual
ITER_D_4N9E8W	Data Visualization Tool User Manual