User Manual

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).

<table>
<thead>
<tr>
<th>Approval Process</th>
<th>Name</th>
<th>Action</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Author</td>
<td>Di Maio F.</td>
<td>08 Feb 2017:signed</td>
<td>IO/DG/COO/SCOD/CSD/CDC</td>
</tr>
<tr>
<td>Co-Authors</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reviewers</td>
<td>Park M.</td>
<td>17 Feb 2017:recommended</td>
<td>IO/DG/COO/SCOD/CSD/CDC</td>
</tr>
<tr>
<td>Approver</td>
<td>Wallander A.</td>
<td>17 Feb 2017:approved</td>
<td>IO/DG/COO/SCOD/CSD</td>
</tr>
</tbody>
</table>

Document Security: Internal Use
RO: Stepanov Denis

Read Access
AD: ITER, AD: External Collaborators, AD: IO_Director-General, AD: EMAB, AD: OBS - Control System Division (CSD) - EXT, AD: OBS - CODAC Section (CDC) - EXT, AD: OBS - CODAC Section (CDC), AD: Auditors, AD: ITER Management Assessor, project administrator, RO, LG: F4E-RH, LG: JADA-RH, GG: STAC Members...
## Change Log

**CODAC Core System Overview (34SDZS)**

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

<table>
<thead>
<tr>
<th>Version</th>
<th>Approved Date</th>
<th>Update Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>v4.0</td>
<td>08 Feb 2013</td>
<td>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.</td>
</tr>
<tr>
<td>v4.1</td>
<td>03 Jul 2013</td>
<td>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.</td>
</tr>
<tr>
<td>v4.2</td>
<td>10 Feb 2014</td>
<td>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)</td>
</tr>
<tr>
<td>v4.3</td>
<td>26 Jun 2014</td>
<td>Minor update for the 4.3.0 release: - Add version 4.3 in history - Expand Maven editor (figure) - Removed VDCT - Confirm &amp; clarify the 1 micro-second accuracy for system time with TCNd - Added PXIe-63868 support - Update list of document (chapter 8)</td>
</tr>
<tr>
<td>v5.0</td>
<td>09 Feb 2015</td>
<td>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.</td>
</tr>
<tr>
<td>v5.1</td>
<td>09 Jul 2015</td>
<td>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 &quot;Public Users&quot; in chapter 2 (distribution suppressed) 4) Replace quarters with months for the planned dates of releases A&amp;B (section 2.4) 5) Add Install/remove commands in the list for ITER Maven tools (section 6.2) 6) Add &quot;Support for Programs&quot; section (7.8) 9) Update the documentation list with the new documents</td>
</tr>
<tr>
<td>v5.2</td>
<td>28 Jan 2016</td>
<td>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)</td>
</tr>
<tr>
<td>Version</td>
<td>Status</td>
<td>Date</td>
</tr>
<tr>
<td>---------</td>
<td>--------</td>
<td>------------</td>
</tr>
</tbody>
</table>
| 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 |
Table of Contents

1 INTRODUCTION ...............................................................................................................3
  1.1 Purpose.............................................................................................................................3
  1.2 PCDH Context.................................................................................................................3
  1.3 Structure of this Document............................................................................................4
  1.4 Reference Documents .....................................................................................................4
  1.5 Acronyms .........................................................................................................................5
  1.6 History..............................................................................................................................5

2 CODAC CORE SYSTEM ..................................................................................................7
  2.1 Overview ..........................................................................................................................7
  2.2 Release Scheme................................................................................................................8

3 USERS, DISTRIBUTION AND SUPPORT.....................................................................9
  3.1 ITER Users ......................................................................................................................9
  3.2 Distribution of the CODAC Core System.....................................................................9
  3.3 User Support....................................................................................................................9

4 SYSTEM COMPONENTS ..............................................................................................10
  4.1 RHEL .............................................................................................................................10
  4.2 EPICS.............................................................................................................................10
  4.3 Other System Components...........................................................................................10

5 SELF-DESCRIPTION DATA TOOLKIT .....................................................................12

6 CODAC SERVICES .......................................................................................................14
  6.1 Mini-CODAC ................................................................................................................14
  6.2 Control System Studio..................................................................................................14
  6.3 Operator Interfaces.......................................................................................................14
  6.4 Alarm Handling ............................................................................................................15
  6.5 Data Archiving ..............................................................................................................16
  6.6 Other CS-Studio Tools .................................................................................................16

7 DEVELOPMENT TOOLS ..............................................................................................17
  7.1 EPICS IOC Development .............................................................................................17
  7.2 ITER Maven Tools........................................................................................................17
  7.3 SNL Sequence Editor .................................................................................................18
  7.4 Support for C/C++, Java and Python .........................................................................19

8 PSH AND CONTROLLERS ...........................................................................................20
  8.1 PLC Integration ............................................................................................................20
  8.2 System Monitoring.......................................................................................................20
  8.3 Timing and Synchronization.......................................................................................21
  8.4 Support for I/O boards...............................................................................................21
  8.5 SDN Support ...............................................................................................................22
1 INTRODUCTION

1.1 Purpose

This document is a public document to provide ITER users with an overview of the software package 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 is 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

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

The PCDH comprises a core document which presents the plant system I&C life cycle and recaps the main rules to be applied to the 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, its context with regards to the Plant Control Design Handbook, as well as lists references, defines abbreviations, and gives the history of how this document has evolved.
- Chapter 2 gives an overview of the CODAC Core System as well as the release scheme for CCS.
- 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 the suite of tools included in the CODAC Core System for working with self-description data.
- Chapter 6 describes the different CODAC services, e.g. alarms, archiving and human machine interfaces.
- Chapter 7 describes the development tools that come packaged in the CODAC Core System.
- Chapter 8 gives some basic information about the controllers contained in a plant system I&C.

Since this document is intended to serve as an introduction to the CODAC Core System, more details are contained in the documentation listed in Chapter 9.

1.4 Reference Documents

[RD1] Plant Control Design Handbook (ITER_D_27LH2V)
https://www.iter.org/org/team/chd/cid/codac

[RD2] EPICS - Experimental Physics and Industrial Control System
www.aps.anl.gov/epics/

[RD3] ITER CODAC Acronyms List (2LT73V)
https://portal.iter.org/departments/CHD/CODAC/Pages/Acronyms.aspx

[RD4] CODAC Core System public web page:
https://www.iter.org/org/team/chd/cid/codac/Pages/CoreSystem.aspx

[RD5] On-line Learning Centre - CODAC Core System Training
https://portal.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 (333J63)
http://static.iter.org/codac/pcdh7/Folder%202/7-ITER_catalogue_for_I%26C_products_-_Slow_c_333J63_v3_3.pdf
### 1.5 Acronyms

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

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA</td>
<td>Channel Access</td>
</tr>
<tr>
<td>COS</td>
<td>Common Operating State</td>
</tr>
<tr>
<td>CS-Studio</td>
<td>Control System Studio</td>
</tr>
<tr>
<td>DAN</td>
<td>Data Archiving Network</td>
</tr>
<tr>
<td>DDD</td>
<td>Design Description Document</td>
</tr>
<tr>
<td>EPICS</td>
<td>Experimental Physics and Industrial Control System</td>
</tr>
<tr>
<td>HMI</td>
<td>Human-Machine Interface</td>
</tr>
<tr>
<td>I&amp;C</td>
<td>Instrumentation and Control</td>
</tr>
<tr>
<td>I/O</td>
<td>Input / Output</td>
</tr>
<tr>
<td>IO</td>
<td>ITER Organization</td>
</tr>
<tr>
<td>MRG-R</td>
<td>Messaging Realtime Grid - Realtime</td>
</tr>
<tr>
<td>NI</td>
<td>National Instruments</td>
</tr>
<tr>
<td>OPI</td>
<td>Operator Interface</td>
</tr>
<tr>
<td>PCI</td>
<td>Peripheral Component Interconnect</td>
</tr>
<tr>
<td>PCIe</td>
<td>PCI Express</td>
</tr>
<tr>
<td>PCDH</td>
<td>Plant Control Design Handbook [RD1]</td>
</tr>
<tr>
<td>PLC</td>
<td>Programmable Logic Controller</td>
</tr>
<tr>
<td>POC</td>
<td>Plant Other Controller</td>
</tr>
<tr>
<td>PON</td>
<td>Plant Operation Network</td>
</tr>
<tr>
<td>PSH</td>
<td>Plant System Host</td>
</tr>
<tr>
<td>RHEL</td>
<td>Red Hat Enterprise Linux</td>
</tr>
<tr>
<td>RO</td>
<td>Responsible Officer</td>
</tr>
<tr>
<td>SDD</td>
<td>Self-description Data</td>
</tr>
<tr>
<td>SDN</td>
<td>Synchronous Databus Network</td>
</tr>
<tr>
<td>SNL</td>
<td>State Notation Language</td>
</tr>
<tr>
<td>TCN</td>
<td>Time Communication Network</td>
</tr>
</tbody>
</table>
1.6 History

- 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.
- 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.

From version 3.0, tools have been included in the distribution to allow the migration from one version to the next one (i.e. migration from 2.x to 3.x or 3.x to 4.x, etc.).
2 CODAC CORE SYSTEM

2.1 Overview

CODAC designates the central control system responsible for operating the ITER device. The different plant systems that constitute the ITER device will each be driven by a local instrumentation and control designated as a plant system I&C. The plant system I&C contains local controllers that are either programmable logic controllers (PLCs) or rack mounted computers controlling PCIe input/output (I/O) and designated as fast controllers.

Each plant system I&C also includes one Plant System Host (PSH), supplied by the ITER Organization (IO) and implements standard functions not requiring plant system specific software.

During development and testing, the CODAC central infrastructure, that will include servers and operator workstations, is replaced with a computer designated as mini-CODAC which implements a subset of the CODAC functions.

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

![Figure 2-1: Plant System I&C Physical Architecture](image)

The 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 mini-CODAC and the PSH and it provides plant system I&C developers with the environment required to develop and test the software in a way that complies with the ITER requirements.
The CODAC Core System comes bundled together with a Red Hat Enterprise Linux (RHEL) Operating System, preconfigured for the CODAC Core System. The CODAC Core System software is based on EPICS [RD2], an open-source control system that is widely used in experimental physics. The software of the PSH and fast controllers is based on the EPICS framework and utilities. The mini-CODAC software is based on EPICS tools, developed within the EPICS community and extended by ITER.

2.2 Release Scheme

All CODAC Core System releases are labelled with a sequence of 3 numbers (the CODAC Core System version identifier) that identify the major version, the minor release, starting from 0 for each major version, and the maintenance release, starting from 0 for each minor release. For example, CCS version 4.3.1 is a maintenance release for the 4.3 version while CCS version 5.1 is a minor release of version 5.

Two official releases have been planned every year until 2016:

- Release “A” in February of either a new major version or a new minor one, introducing major changes and new features
- Release “B” in June or July of a new minor version, improving and extending the previously released version. Alternatively, a maintenance version can be released if only corrections and minor enhancements are introduced without new features

From 2017, only one version a year will be planned (release A in February) In addition, maintenance versions may be released if bug fixes need to be distributed to many users.

Patches can also be distributed to a limited number of users for fixing one or a few bugs.

Beta releases are produced before any official versions but these can only be distributed to testers and shall not be used for I&C development.

Two or more different versions can be installed on one computer, provided the RHEL version is the same, but only one version can be activated at a time. This feature is provided mainly to facilitate migration to the latest version.

It is strongly recommended to migrate to the latest version of the IO CCS distribution. Tools are included for upgrading systems and for migrating applications. The support team can also assist users to facilitate such a migration.

With each release, the list of enhancements and bug fixes is recorded. This list is published via the release notes and the details can be obtained from CODAC support. There is no retrofitting of bug fixes to previously released versions but patches can be provided, if required.
3 USERS, DISTRIBUTION AND SUPPORT

3.1 ITER Users

Users who contribute to the development of the ITER I&C System, such as ITER Domestic Agencies, institutes associated with them or industry working for ITER through contracts, may become registered ITER community users, or in short, registered users.

The procedure for becoming a registered user is described in the ITER CODAC public web pages [RD4] and consists of completing and submitting a form which includes the name of an IO contact person and a justification. Requests are handled by the IO CODAC Responsible Officer.

Registered users have access to all official ITER software releases and to full support services.

3.2 Distribution of the CODAC Core System

The distribution of the CCS software is implemented through a server at IO premises that provides users with all the required software packages (RPM packages on a YUM repository) for configuring their computers. Each computer on which the software shall be installed has to be registered.

Different profiles are provided in order to allow configuring the different systems: mini-CODAC, PSH, fast controller or CODAC terminal. These profiles are available in two variants, one for development that includes the development tools and one for operation that doesn’t.

The documentation (listed in chapter 9) is included in the software distribution.

The examples that are provided as part of the user documentation for a component (e.g., PLC driver, PXI-6259 EPICS interface, etc.) are also included in the distribution.

3.3 User Support

A support service is available for all registered users. The support is available by e-mail and the service operates continuously during IO opening hours.

An issue handling system has been set-up to keep track of reported bugs, known issues or planned extensions. This is the main tool for the follow-up of the enhancements and bug fixes, and for the planning of the releases.

Hands-on workshops are organized in Cadarache, on a regular basis, for ITER I&C developers as well as dedicated sessions at Domestic Agencies premises according to needs. On-line training is also available for ITER users on the ITER On-line training center site [RD5].

Since the release of the version 5.0, the support for the 4.x versions is maintained but the support for 3.x versions will be limited to assistance for migration and no regular support is provided for 2.x and 1.x versions.
4 SYSTEM COMPONENTS

4.1 RHEL

Red Hat Enterprise Linux (RHEL) is the standard operating system for mini-CODAC, the PSH and the fast controllers. An officially supported version of RHEL is required for these systems. The operating system is included in the distribution and the first part of the installation will install the operating system on any target computer.

The supported architecture is linux-x86_64.

RHEL 6 was introduced with CCS version 3 (RHEL 6.1 in version 3.x), RHEL 6.3 in CCS version 4.x and RHEL 6.5 from CCS version 5.0.

The real-time enabled CCS version, MRG-Realtime (MRG-R), is also supported for fast controllers: MRG-R 2.0 in CCS version 3.x, MRG-R 2.1 from CCS version 4.0 and MRG-R 2.5 from CCS version 5.0.

4.2 EPICS

The EPICS base is included in the CCS distribution and is required for all system profiles. The EPICS framework is the base for the fast controllers and the PSH, and the EPICS communication protocol, Channel Access (CA) which is the standard communication protocol for access to plant system I&C over the Plant Operation Network (PON).

EPICS base version 3.14.12 was introduced with CCS version 2.

EPICS base version 3.15 was introduced with CCS version 5.

The following major EPICS components are included:

- State Notation Language (SNL), for the development of state machines in PSH, fast controllers and mini-CODAC.
- Autosave for saving/reloading the configuration values to/from disk when persistency is required.
- The standard EPICS IOC error logging, for error and trace logging.

In addition, device supports for the implementation of the EPICS interface with external non-EPICS systems are provided:

- the asyn module for interfacing via a client library
- the streamDevice module for interfacing via commands/reply protocols
- the areaDetector module for image acquisition systems
- an OPC-UA device support for interfacing via the OPC-UA protocol

4.3 Other System Components

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

- A relational database, built with PostgreSQL, for storing the plant system I&C data for development on mini-CODAC as well as for the archives, alarms and operator logs for operation.
■ Eclipse, with plug-ins for C/C++, Java, Python, Subversion, and with all required plug-ins for Control System Studio.
■ Apache ActiveMQ as the JMS implementation, as required for Control System Studio
■ The Spring Framework and Hibernate, as required for self-description data tools.
■ The Apache Tomcat servlet container for web applications.
5 SELF-DESCRIPTION DATA TOOLKIT

Self-Description Data (SDD) is an ITER concept designating 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, on a local development system, such as a mini-CODAC system.

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

This toolkit now 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 into the required configuration data for mini-CODAC, the PSH and the fast controllers and into the required STEP-7 files for interfacing PLCs.
- The **SDD sync** tool to save and load SDD data to/from XML files and to synchronize local SDD databases with IO databases and local files with the IO source repository.
- The **SDD parser** to parse user-provided or user-modified configuration files, such as EPICS record definition, and retrofit the changes into the SDD database.
- The **SDD web application** can be used via a web browser and can operate on remote SDD databases for browsing I&C configurations, transferring SDD data between local and central data bases or generating configuration files.

![Figure 5-1 - SDD Editor: List of variables in a function of a plant system I&C.](image-url)
The SDD tools provide the user with features to create, edit and save:

- The list of signals interfaced by the plant system I&C
- The list of functions and variables implemented by the plant system I&C
- The list of control units (PSH, controllers) that belong to the plant system I&C
- The communication between PSH and PLC
- The configuration for alarms
- The configuration for archiving
- The configuration for operator interfaces (OPIs)
- The configuration for the supported I/O boards
- The cubicles that shall be monitored
- The mapping of Common Operating State (COS) variables into plant-system specific variables
- The SDN (section 8.5) data produced or consumed by programs running on fast controllers
- The DAN (section 8.6) data produced by programs running on fast controllers (from CCS5.0)
- The location of cubicles and position of controllers and chassis in the cubicles (from CCS5.0)
- The association between EPICS variables and programs (from CCS5.0)

The project structure allows creating I&C projects that correspond to a single delivery (e.g. one Procurement Arrangement). An I&C project can cover more than one plant system I&C.

The SDD tools enforce the usage of the Control Breakdown Structure (CBS) defined in the CODAC Design Description Document (DDD) [RD5] and are configured with the official nodes of the 1st level.
6 CODAC SERVICES

This chapter briefly introduces the services of the CODAC Core System for operation.

6.1 Mini-CODAC

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

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

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

For this purpose, a version of the operational application has been included from CCS5.2:

- configuration editor
- supervision
- test tools for SDN data exchange

After integration, the CODAC services will be provided from CODAC servers and the development tools shall be used from on-site 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

Eclipse has been adopted as the development platform for the mini-CODAC software.

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 archives.

6.3 Operator Interfaces

The display manager, BOY, developed with the CS-Studio platform, allows the development of dynamic displays by means of 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 future requirements.

A dedicated library of industrial symbols, with graphical widgets displaying these, and with dynamic attributes, has been added for ITER applications (I&C OPIs).
BOY is being continuously extended by its users within a collaboration. New features are added in each release. Extensions are listed in the release notes.

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

![Figure 6-1 - OPI compliant with ITER HMI](image)

It is also possible to run BOY displays in a web browser.

### 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 such as sending an email or submitting a new logbook entry.

Alarm logs and configurations are stored in the mini-CODAC relational database.

Figure 6-2 is an illustration of the alarm display.
Alarm descriptions can be announced by a text-to-speech converter. It is also possible to monitor alarms in a web browser.

6.5 Data Archiving

The EPICS data archiver, BEAUTY, is also included in the CCS distribution. It allows deploying archive engines that save data into a relational database on the mini-CODAC system.

A data viewer, the CS-Studio DataBrowser, allows retrieval of archived data as well as monitoring of live data.

Data archiving requires that the set of variables to be stored is configured with SDD tools. It is also possible to visualize archived data in a web browser.

6.6 Other CS-Studio Tools

Additional CS-Studio tools have been included in the CCS distribution:

- An electronic logbook (ELOG)
- A tool for creating and executing sequences of commands (Scan server)
- Tools for inspecting EPICS variables (PV Manager Probe and PV Fields Viewer)
7 DEVELOPMENT TOOLS

7.1 EPICS IOC Development

EPICS includes a framework for developing control processes, called IOC processes. IOC (Input Output Controller) designates controllers in the EPICS community. Each IOC process is built as a database of records that belong to a set of pre-defined record types.

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

7.2 ITER Maven Tools

I&C projects are developed using an ITER specific workflow that is supported by the SDD tools and by commands implemented using the Apache Maven build tool.

The development workflow is composed of the following steps:

1. Editing of the I&C project with the SDD Editor and/or web application.
2. Generation of the EPICS/CS-Studio/STEP-7 configuration files with the SDD translator.
3. Creation/update of the software unit with dedicated commands.
4. Editing of user-defined files with a text editor or specific editors, such as the CS-Studio SNL editor and development of C++ programs if any.
5. Compilation of the EPICS IOC processes and of the C++ programs.
6. Testing the project: start, stop, status and test commands.
7. Creation of the software packages for distribution.
8. Installation and removal of the software packages.

Maven commands (checkout, commit, import, update) also allow saving, restoring and synchronizing the project’s files with a Subversion (SVN) source repository, either the ITER central repository or a local one during off-line development phases.

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 SDD editor.

As illustrated in Figure 7-1, from the Maven Editor interfaces, the user can build the software, run the executables for test purposes, on local or remote hosts, and build the packages for deployment.
7.3 SNL Sequence Editor

![SNL Sequence Editor](image)

Figure 7-2 – SNL Sequence Editor, shown in CS-Studio, is also available in SDD Editor

Add new EPICS application
Stop, Clean, Compile and Run
IOCs status...
Run IOCs...
Connect to console
Stop IOCs...

Figure 7-1 - Maven Editor – EPICS commands
The **SNL Sequence Editor**, a standard EPICS tool, is included to allow the development of EPICS state machines using the EPICS state notation language.

A language sensitive editor as well as the environment for testing the sequence and building a graphical representation of a state diagram is included in SDD editor and CS-Studio (Figure 7-2).

SNL programs can be developed and added to EPICS IOC processes for implementing state machines, such as plan-system operating state (PSOS) management on a fast controller.

### 7.4 Support for C/C++, Java and Python

C and C++ are the programming languages used for controller-level software and for real-time applications, Java is the main language for mini-CODAC software and Python is the recommended language for rapid prototyping and for testing.

For each of these languages, a Channel Access client library is provided as well as Eclipse configurations for development.
8 PSH AND CONTROLLERS

This chapter describes some of the CODAC Core System features related to the controllers contained in a plant system I&C.

8.1 PLC Integration

PLCs used in plant system I&Cs shall be SIMATIC S7 PLCs and these are programmed using the Siemens development tool, SIMATIC STEP 7. The recommended components are described in the ITER catalogue for I&C products - Slow controllers PLC [RD7].

The CODAC Core System distribution includes 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 an EPICS IOC process deployed by default on the PSH, alternatively on a fast controller.

The communication consists of 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 into an EPICS process variable in the PSH and into a program variable in the PLC.

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

Redundant PLCs (e.g. interlock systems) are supported. This includes the transparent connection to the active master CPU and the reconnection to the new one in the case that a switch is triggered because of a failure.

8.2 System Monitoring

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

Each IOC process is built with a monitoring module that provides, by means of automatically generated variables, information about the process (process ID, start time, port used…) and its resource usage (CPU, memory, registered CA clients …).

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

Diagnostic displays using these variables can be developed or generated in an automatic 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), as implemented by dedicated PLCs in standard cubicles, is also interfaced via such variables.
8.3 Timing and Synchronization

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

PXI boards compliant with this protocol are supported for fast controllers as well as, from CCS version 5.0, PTP compliant Ethernet interfaces of the computers.

The timing and synchronization board for fast controllers is **NI PXI-6683H**. Support for the older version, **NI PXI-6682**, is limited.

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.

8.4 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 PCDH [RD1]. The scope of the software support is limited to the I/O boards that are defined in this catalog and that 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 now supported.

These are:

- A multi-function data acquisition board, **NI PXI-6259**, with the following channels:
  - 16/32 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 counters (32 bits)
- 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.
- The **NI PXIe-6368** board (X-series), which is a high performance (2 MS/s simultaneous) multi-function data acquisition board.
- The **NI CompactRIO** modules described in the Fast Controllers catalog.
8.5 SDN Support

The Synchronous Databus Network (SDN) is the high performance real time network that will be used for real time data sharing 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 and the programs deployed on fast controllers can be configured for publishing data on SDN topics or for subscribing to these.

The SDN Application Programming Interface (API) is included in the CCS distribution. This is a library for C++ programs that implements all the required services for SDN communications. An SDN monitoring services can also be started for gathering information on SDN data exchanges.

8.6 DAN Support

The Data Archiving Network (DAN) is the dedicated network for streaming high throughput 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 such data.

The 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) for allowing processes that run on fast controllers to publish data on DAN.

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.

8.7 Common Operating State Support

COS variables are a mandatory part of the plant system I&C interface with CODAC. It provides the supervision 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 and the implementation is configured by means of a mapping table between the generic COS variables and the specific ones that the I&C developer should implement. These specific variables can be implemented either on a fast controller or on a PLC.
8.8 Support for Programs

In addition to EPICS IOCs, programs can be added to the configuration of a fast controller using SDD tools. For each program, in addition to the SDN and DAN configurations, the EPICS interface as well as the I/O boards definition can be defined for generating the appropriate code for implementing the CODAC interface or for using an I/O board.
9 DOCUMENTATION

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

**Overview**

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_34SDZ5</td>
<td>CODAC Core System Overview</td>
<td>(This document)</td>
</tr>
<tr>
<td>ITER_D_43PSH9</td>
<td>CODAC Core System User Manual</td>
<td></td>
</tr>
</tbody>
</table>

**Installation & Support**

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_33JNKW</td>
<td>CODAC Core System Installation Manual</td>
<td></td>
</tr>
<tr>
<td>ITER_D_PKHN4H</td>
<td>CODAC Installation Manager User Manual</td>
<td></td>
</tr>
<tr>
<td>ITER_D_7JCFUD</td>
<td>CODAC Core System Migration Guide</td>
<td></td>
</tr>
<tr>
<td>ITER_D_U2TTSZ</td>
<td>PTPd User Manual</td>
<td>(new)</td>
</tr>
</tbody>
</table>

**Development & Testing**

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_33T8LW</td>
<td>CODAC Core System Application Development Manual</td>
<td></td>
</tr>
<tr>
<td>ITER_D_32Z4W2</td>
<td>SDD Editor User Manual</td>
<td></td>
</tr>
<tr>
<td>ITER_D_42NFCY</td>
<td>SDD Web Application User Guide</td>
<td></td>
</tr>
<tr>
<td>ITER_D_46AAXR</td>
<td>SDD Synchronization Guide</td>
<td></td>
</tr>
<tr>
<td>ITER_D_A4WQDZ</td>
<td>How to include a new I/O module in SDD</td>
<td></td>
</tr>
<tr>
<td>ITER_D_7MT2YC</td>
<td>Maven Editor User Guide</td>
<td></td>
</tr>
<tr>
<td>ITER_D_35XFCY</td>
<td>System Health Monitoring Variables</td>
<td></td>
</tr>
<tr>
<td>ITER_D_QEK784</td>
<td>Logging library - Software User Manual</td>
<td></td>
</tr>
</tbody>
</table>

**Control System Studio**

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_3XLESZ</td>
<td>HMI Style Guide and Toolkit</td>
<td></td>
</tr>
<tr>
<td>ITER_D_7367JQ</td>
<td>Operator Interface standardisation - CSS BOY Edition and Runtime</td>
<td></td>
</tr>
<tr>
<td>ITER_D_A69URK</td>
<td>Operator Interface standardisation - CSS BOY Industrial Symbol Library</td>
<td></td>
</tr>
</tbody>
</table>

**Slow Controllers (PLC)**

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_6KFJEJ</td>
<td>S7 PLC EPICS Driver Extensions</td>
<td></td>
</tr>
<tr>
<td>File Path</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>ITER_D_2N8C3M</td>
<td>PLC Sample Application Guide</td>
<td></td>
</tr>
<tr>
<td>ITER_D_PJAHXJ</td>
<td>s7PLCA syn EPICS Driver User's Manual</td>
<td></td>
</tr>
</tbody>
</table>

### Fast Controllers - IO

<table>
<thead>
<tr>
<th>File Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_2PLQ4P</td>
<td>NI Sync Linux Device Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_33Q5TX</td>
<td>NI Sync EPICS Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_32GTJY</td>
<td>NI PXI-6259 Linux Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_3DEY52</td>
<td>NI PXI-6259 EPICS Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_3ZHXQ9</td>
<td>NI PXI-6528 Linux Device Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_433VEW</td>
<td>NI PXI-6528 EPICS Driver User’s Guide</td>
</tr>
<tr>
<td>ITER_D_LW3UHF</td>
<td>NI-RIO Device Driver User Manual</td>
</tr>
<tr>
<td>ITER_D_RAJ9P8</td>
<td>NI-RIO EPICS Device Driver User Manual</td>
</tr>
<tr>
<td>ITER_D_3LTMR6</td>
<td>NI X-Series Linux Device Driver User's Guide</td>
</tr>
<tr>
<td>ITER_D_3P4N3R</td>
<td>NI X-Series EPICS Driver User's Guide</td>
</tr>
<tr>
<td>ITER_D_A6LWQ8</td>
<td>NDS User Manual</td>
</tr>
</tbody>
</table>

### Fast Controllers - HPN

<table>
<thead>
<tr>
<th>File Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_N4XTGG</td>
<td>TCN API - Software User Manual</td>
</tr>
<tr>
<td>ITER_D_MUYNT6</td>
<td>TCNd User Manual</td>
</tr>
<tr>
<td>ITER_D_B7SKFU</td>
<td>SDN Software User Manual</td>
</tr>
<tr>
<td>ITER_D_Q6GULS</td>
<td>DAN User Manual</td>
</tr>
<tr>
<td>ITER_D_SD29MG</td>
<td>SDN archiver user manual</td>
</tr>
</tbody>
</table>

### Operational Applications

<table>
<thead>
<tr>
<th>File Path</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITER_D_RUZCJ9</td>
<td>User Manual for CODAC Operational Applications</td>
</tr>
<tr>
<td>ITER_D_RGSWH</td>
<td>SDN Tools User Manual</td>
</tr>
</tbody>
</table>