# General P25 Wireline Overview

The importance of the Project 25 (P25) suite of standards has been conveyed by congress through several pieces of legislation. With congress providing grants to all levels of government (local, state, and federal) for the acquisition of public safety telecommunications equipment the suite of P25 standards is necessary to ensure interoperability amongst the different levels of government and between agencies at the same level of government. In terms of completion, the ISSI interface happens to be the interface congress considers top priority; hence, this is why the set of ISSI test tools is being developed first. The ISSI, CSSI, and FSI are three interfaces out of a total of seven P25 interfaces.

**Specific P25 Wireline Project Information**

The purpose of this project is to develop a set of open source test tools to assist with conformance, performance, & interoperability testing of the Project 25 wireline interfaces. Wireline interfaces in this context refers to interfaces that transport information via UDP/IP.  Test tools for P25 wireline interfaces developed by commercial vendors currently do not exist. Test tools development is based on the test documents created by TIA TR-8. The three primary interfaces being focused on are:

* ISSI - Inter-RF Sub-System Interface
* CSSI - Console Sub-System Interface
* FSI - Fixed Station Interface

Test tools are being developed for all three of these interfaces based on the following documents.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **P25 Wireline Interface** | **Protocol Specification** | **Conformance Specification** | **Performance Measurement Methods** | **Interoperability Specification** |
| ISSI | TIA-102.BACA-A | TIA-102.BAC?\* | TIA-102.CACA | TIA-102.????\* |
| CSSI | TIA-102.BACA-2 | TIA-102.BA??\* | TIA-102.CACA-1 | TIA-102.????\* |
| FSI | TIA-102.BAHA-A | TIA-102.CADA | TIA-102.????\* | TIA-102.????\* |

\* Indicates that the specification is currently under development

For the current state of the above specifications, please refer to [Project 25 Documents & Standards](http://www.pscr.gov/outreach/p25dsr/menu_top/p25_documents_quick_status.php)

The ISSI and CSSI projects under the p25-wireline rely heavily on the following java.net VoIP projects:

* [JRTP](http://java.net/projects/jrtp)
* [JAIN-SIP](http://java.net/projects/jain-sip)

ISSI stands for Inter-RF Sub-System Interface, and it can essentially be thought of as an IP based network protocol that is able to utilize a standard COTS network interface card (NIC). The intent is for this interface to be present in future deployments of P25 Radio Frequency Sub-Systems (RFSS). The ISSI of different RFSS’s can be inter-connected using various medium (for example Ethernet). When the ISSI’s of various RFSSs are interconnect this allows IP packets that contain encoded voice to be transmitted and received. The ISSI is important to public safety agencies because it will foster competition amongst several vendors who manufacture RFSSs. This increased competition will essentially drive down the cost of P25 infrastructure over time. In addition to lower cost, the most important reason for having an ISSI interface is to promote interoperability between the different vendors who manufacture RFSSs. This allows the end consumer to implement a P25 communications network that consists of RFSSs from multiple vendors.

**ITT Downloads**

Please read the p25-wireline license information before downloading.  For the latest build and source code please check the Subversion page.  For the latest bug report information please check the Issue Tracker page.  Additional information can be found on the Documents and files page.

To download, please access the **Downloads** link under the Project Features

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Release** | **Date** | **Source** | **Binary** | **Note** |
| ITT-1.0.RC2 | 08/14/2008 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC2/issi-tools-1.0.rc2.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC2/itt-install-1.0.rc2.jar) | ITT baseline release |
| ITT-1.0.RC3 | 10/02/2008 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC3/issi-tools-1.0.rc3.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC3/itt-install-1.0.rc3.jar) |  |
| ITT-1.0.RC4 | 01/29/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC4/issi-tools-1.0.rc4.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC4/itt-install-1.0.rc4.jar) |  |
| ITT-1.0.RC5 | 03/24/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC5/issi-tools-1.0.rc5.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC5/itt-install-1.0.rc5.jar) |  |
| ITT-1.0.RC6 | 07/13/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC6/issi-tools-1.0.rc6.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC6/itt-install-1.0.rc6.jar) |  |
| ITT-1.0.RC7 | 09/16/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC7/issi-tools-1.0.rc7.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC7/itt-install-1.0.rc7.jar) |  |
| ITT-1.0.RC8 | 10/25/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC8/issi-tools-1.0.rc8.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC8/itt-install-1.0.rc8.jar) |  |
| ITT-1.0.RC9 | 12/01/2009 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC9/issi-tools-1.0.rc9.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC9/itt-install-1.0.rc9.jar) | This version allows multiple instances of ITT to  run on a single PC in order to test an implementation. |
| ITT-1.0.RC10 | 12/15/2010 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC10/issi-tools-1.0.rc10.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.0-RC10/itt-install-1.0.rc10.jar) | This version contains fixes for close to 85% of bug  reports contained in the issue tracker database. |
| ITT-1.1 | 02/03/2012 | [zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.1/issi-tools-1.1.zip) [p25stack-1.0-zip](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.1/p25stack-1.0-src.zip) | [binary](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.1/itt-install-1.1.jar) [p25stack-1.0-bin](http://java.net/projects/p25-wireline/downloads/download/ITT-V1.1/p25stack-1.0.jar) | This version contains a standalone p25stack library. ITT has been tested against real RFSSes and in a VM  environment.  As of 02/03/2012, the project is completed. |

**Note**:  After downloading the *binary* file on a machine running Windows XP, the Windows machine may recognize the file as a .zip rather than a .jar.  For java to execute the binary the file must have a .jar extension.  The file must be renamed through the DOS command line.  The command that needs to be issued will be similar to the following:

"rename itt-install-1.0.rc3.zip itt-install-1.0.rc3.jar"

 The file name prefix will vary depending on the release of the software you are using.

**Project Milestones**

Some of the milestones for the ITT project are as shown below.

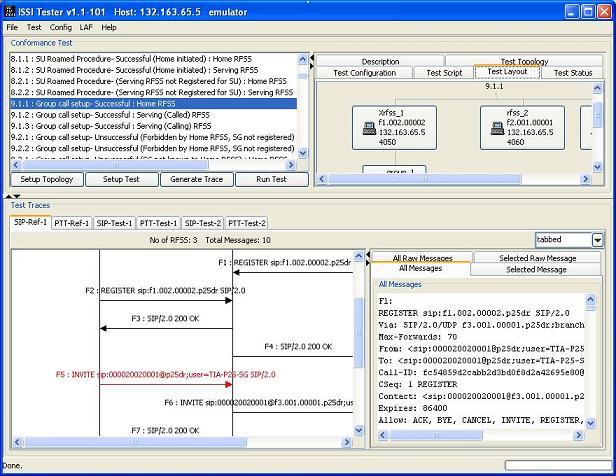
|  |  |  |
| --- | --- | --- |
| **Task** | **Description** | **Est Completion Date** |
| Conformance Test Case Verification | Verify/validate conformance test cases against the ISSI conformance test document | Nov 2008 |
| RC1 GUI Overhaul | Enhance current GUI for usability. | mid-Sept 2008 |
| SIP Header Formatting | Refactor & clean up SIP Protocol header; Modify the SGID field in SIP header to conform to ISSI specification | end-Sept 2008 |
| Enforcement of Java Coding Standards | Remove excess usage of keyword "this"; modify code to use getters & setters; develop code to propagate and handle exceptions throughout ITT | early-Oct 2008 |
| Java Unit Testing | Verify that unit test cases are working properly; verify that ant targets in build.xml fiel are valid and modify as appropriate | mid-Oct 2008 |
| Trace Analyzer Modifications | Consolidate system config files; decouple system config files from trace analyzer GUI; display hex dump info for selected message; remove hex dump display for all messages; Sub-system interop verification | end-Oct 2008 |
| Documentation | Quick start guide describing conformance test tool operation and that also describes how to use the trace analyzer | end-Oct 2008 |
| Performance Measurement Verification | Verify clock drift of server before NTP applies correction; verify current code performs timing measurements properly; Verify that all ISSI performance tests have been implemented from MM doc; verify performance results returned by ITT are valid for group call setup | mid-Nov 2008 |
| Test ITT against live RFSSs. | In the NIST PSCR lab ITT was validated against live RFSSs from Raytheon & Cassidian (formerly known as EADS). | Jan 2012 |
| Test ITT in a VM Environment | ITT was tested in a virtual machine environment using VMware Workstation.  VMware Workstation was loaded on a sever that had Ubuntu as its OS.  Five Windows XP VMs were created.  ITT was installed in each VM. | Jan 2012 |

**ITT Conformance Tester**

The ISSI conformance test case document is currently being developed by TIA TR-8, and it is expect that a formal published version of the ISSI conformance test document will be available in early 2009. These conformance tests will verify that the vendor implementation under test conforms at a message level to what has been specified in TIA-102.BACA-A.  In order to objectively verify that a vendor conforms to TIA-102.BACA-A a reference implementation of the ISSI protocol stack has been developed.  This software reference implementation is referred to as ISSI Test Tools (ITT).  The development of this software was funded by NIST Public Safety Communication Research (PSCR) Program (www.pscr.gov).

Since ITT is implemented in Java, the software can be loaded on a regular desktop PC that has a Linux or Windows operating system on it.  The roles ITT can emulate are calling serving RFSS, calling home RFSS, called home RFSS, and called serving RFSS.  The preferred configuration when attempting to determine a particular vendor’s conformance is to test a single vendor's ISSI in isolation.  This scenario implies that there is only one vendor RFSS with a real ISSI and the rest of the ISSI interfaces are emulated by ITT.  The number of emulated ISSIs and the role of the emulated interfaces will vary depending on the conformance test case under consideration.  Since ITT does not currently have the capability to emulate the P25 common air interface, the behaviors of (or events generated by) subscriber units are emulated in the ITT software.  The conformance tests are implemented in XML scripts.  The user has the capability to modify the specific test parameters in these scripts as necessary.  From the ITT GUI, the user selects a conformance test case to execute.  After the test case has completed execution, the user can then view the SIP and RTP Push-to-Talk (PTT) messages that were exchanged between ISSIs in a graphical message sequence chart (MSC).  Upon completion of test case execution, ITT will automatically declare either pass or fail.  Raw IP packet data can be rendered by clicking on the message of interest in the MSC.

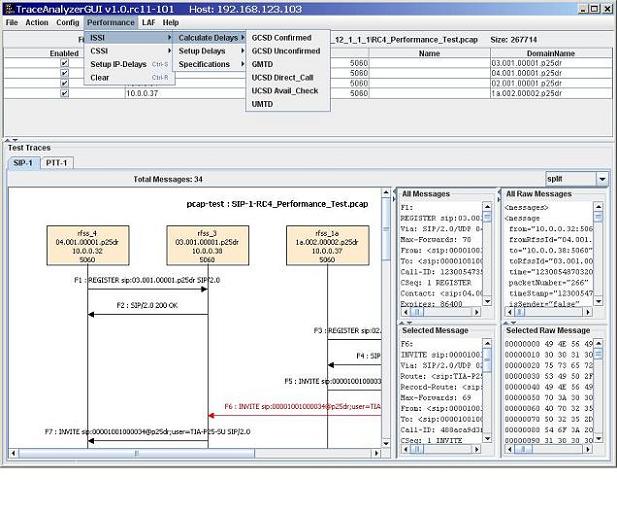
A sample screen of the ISSI Conformance Tester is shown below.



**ITT Trace Analyzer**

The ITT has a trace analyzer ability that can process a PCAP file from [Wireshark](http://www.wireshark.org/) and render a message sequence chart (MSC). This gives ITT the capability to analyze the messaging that occurs between all ISSIs (emulated or real) from performance perspective.

A sample screen of the Trace Analyzer is shown below.



- See more at: https://p25-wireline.java.net/#sthash.sxkNVvdD.dpuf