METEORCOMM LLC

ITCC Integration Test Best Practices Guide

Document Revision: 1.0
Document Number: 00002483-A

Document Number: 00002483-A

Revision History

Revision	Date	Summary of Changes
1.0	10/18/2012	First draft of document for FRA grant.

Third party trademarks

- HP Quality Center and HPQC are trademarks of HP and its suppliers or licensors.
- Linux® is the registered trademark of Linus Torvalds in the United States and other countries
- Microsoft®, Microsoft Excel®, and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Python is a registered trademark of the Python Software Foundation.
- Red Hat®, Enterprise Linux®, Red Hat® Enterprise Linux® (RHEL) are registered trademarks of Red Hat, Inc.
- Wi-Fi is a trademark of the Wi-Fi Alliance.

Executive summary

The Positive Train Control (PTC) communication system program includes hardware and software development, testing, and integration activities that lead to system validation testing in both an integrated lab and closed track environment. This integrated test activity is necessary as an end to the subsystem development cycle to combine the Radio, Messaging, and Systems Management subsystems into an integrated PTC communication system for validating system-level functionality and performance. Much of what is discovered in integration test will directly inform the use of the PTC communication system during and after deployment in the real world.

Conceptually speaking, the integration and testing of the PTC communication system in a lab environment is a prerequisite to field testing. As such, the primary objective of integration lab testing (ILT) is to validate that the PTC communication system, with all subsystems combined, functions in an acceptable manner, the metrics for which are identified in the Interoperable Train Control (ITC) requirements per and built upon each major product release cycle.

The scenarios and test cases for ILT focus on system-level behaviors that encompass multiple subsystems of the PTC communication system. To achieve a positive outcome for a given end-to-end test scenario, all associated elements of the communication system need to work correctly. In the process of executing system-level end-to-end type tests, these elements are therefore indirectly validated. A basic assumption for ILT is that the Messaging software, Systems Management software, and 220 MHz Radio subsystems have undergone extensive testing prior to the start of integration. This means that a large percentage of the ITC requirements are validated at the subsystem level. Therefore, direct revalidation of the functionality and performance of these individual subsystems during integrated system testing is minimized.

After initial verification and testing in a lab setup, integrated test activities are also undertaken in the field. Closed track testing (CTT) of the PTC communication system entails the installation and operation of the communication system at a facility that houses both railroad and communications-related equipment such as locomotives, base towers, wayside poles, and non-revenue track. The facility allows the communication system to be tested under controlled field conditions using real locomotives at operating speeds in excess of 100MPH.

The installation of the PTC communication system in an integrated setup includes various system integration test beds in a lab environment and the CTT field setup noted above. Typically, these test beds include 1-to-3 Base radio installations, 3-to-15 Wayside communication site locations, 2-to-6 Locomotive sites (with multiple sites in real locomotives at CTT), and simulated back offices, which may be local or remote as well as home or foreign. All endpoint applications including that of wayside interface units, train management computers, and back offices are simulated.

Both ILT and CTT start with the basic validation of system-level elements such as the 220 MHz Radio network, remote and back office Messaging system, and Systems Management agents. Once the basic elements have been established as operational, a variety of scenarios are tested. These focus on message delivery between endpoints and typically include multiple back offices, alternate messaging paths, mobility, direct and virtual wayside status, and a variety of failure scenarios.

This document includes detailed descriptions and best practices of various integration test setups and processes such as test case development, test environment setup, and product defect management. Test cases are created from one or more ITC requirements or derived requirements, which stem from use cases perceived to be of high importance or a high probability of occurring in real-world operation. Test cases cover nominal operation as well as failure modes and key performance metrics.

DCN 00002483-A

Table of Contents

Table of Figures

Table of Tables

1. Introduction

1.1 Document purpose

The purpose of this document is to provide a record of the integration test activities and best practices associated with Meteorcomm's (MCC) development of the PTC communications system. These activities cover system-level tests in both lab and field environments of an integrated PTC communication system that is comprised of the PTC 220 MHz Radio (ITCR), Messaging software (ITCM), Systems Management software (ITCSM) subsystems, and includes end-to-end (ETE) communication through the system's various protocols and transports. MCC's approach to integration including test strategy, validation processes, product defecting, test setup and control, system scenarios, and test case development are described within this document.

A communications integration framework and associated test plans have been built up and executed upon. Moreover, the ongoing development of both the PTC communications system and MCC's integration test processes is sustained by both lab and field test environments capable of supporting the testing of current and future PTC components and technologies. The ongoing collection and subsequent analysis of test results will continue to refine and expand test cases and interoperability use cases. In summary, employing these best practices will reduce the risk associated with equipment and software non-compatibility as deployment and operation of the PTC communication systems gets fully underway.

1.2 Assumptions

The MCC approach to its integration test activities are in part formed by the following assumptions:

- MCC is responsible for validating message delivery to end points, but not for validating end point functionality or performance in response to messages.
- Integration test is contingent on the subsystem workstreams providing completely tested SW and HW, minus requirements that require an integrated system. As such, prior to both ILT and CTT, subsystems and their associated SW and HW components have been tested and verified.

- Integration test uses unique tools and test cases from that of (and therefore does not rely on the tools of) the individual subsystem workstreams.
- MCC resources including test control and tracking methods as well as the necessary contracts and SOWs with third party vendors are in place well in advance of and throughout integration test activities.

1.3 Scope

The PTC communication system program includes HW and SW development, testing, and integration activities. MCC's integration activities cover system-level tests for system-level validation and characterization of an integrated PTC communication system that is comprised of its various subsystems. Integration testing is not intended to revalidate features and functionality tested at the subsystem level. Rather, using system-level test scenarios and appropriate key performance indicators (KPIs), integration testing is used to evaluate communication system performance in both a lab and a controlled field environment. In support of this, the following guidelines and principles have been applied and adhered to during MCC's integration activities:

- Scenarios and test cases for integration testing focus on system-level behaviors that encompass multiple subsystems of the PTC communication system.
- For a given ETE test scenario, many elements of the communication system need to work correctly. As such, during ETE testing, these elements are indirectly tested.
- Per each release, each subsystem (ITCM, ITCSM, and ITCR) has undergone extensive testing prior to the start of integration testing. Direct revalidation of the functionality and performance of these individual subsystems during integration testing is minimized.

More specifically, integration activities largely focus on the following test conditions and scenarios (more details can be found elsewhere in the document):

- Where at least two subsystems are operating together (that is, covering the integration "touch points"):
  - Full Integration: ITCR-ITCM-ITCSM
  - Subsystem Pair testing: ITCR-ITCM, ITCR-ITCSM, ITCM-ITCSM

- At various levels of testing, such as:
  - Basic Functionality: validation of basic system operation
  - Stability Testing: confidence testing including soak tests
  - Regression Testing: testing of system functionality from previous releases
  - New Functionality Testing: testing of system functionality including negative test scenarios from the most recent release
  - System-Level KPI Testing: baselining of system performance such as message success, latency, and data throughput
- Inbound/Outbound messaging between all end points, which includes re-routing messages over various transports, fragmenting of large messages, direct peer-to-peer messaging such as wayside status to oncoming locomotives, and interoperability between home and foreign assets
- Mobility such as message delivery during locomotive handoff as well as other RF-link testing such as message delivery with or without base coverage and/or alternate IP paths
- Systems Management functionality such as remote upgrades of ITCR and ITCM assets via kit distribution and remote file activation

As further clarification, some examples of items that are explicitly out of scope for integration testing include the following:

- Real endpoints (TMCs, WIUs, or BOs): instead these are simulated using integration reference architecture
- Testing that includes MPLS connections: no IP impairment testing
- Validation of the over-the-air (OTA) facets pertaining to IP network performance such as Cell or Wi-Fi

1.4 Acronyms, terms, and definitions

Table 1 contains a list of acronyms used in this document and their meaning. Table 2 contains definitions for various terms used in this document. The purpose of these tables is to improve clarity and provide a reference for the reader. The reader is encouraged to refer back to these tables as often as needed.

Table 1: Acronyms and abbreviated terms

Acronym	Description
220	Radio transport or actual 220 MHz frequency band, depending on context
ACK	Acknowledge[d]
AG	Application Gateway
AM	Asset Manager (this specific term has been replaced by “gateway”)
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
AR	Asset Registry
ASA	Asset SMA Adapter
B	Base Radio
BCT	Backward Compatibility Testing
BER	Bit Error Rate
BO	Back Office
Cell	One of the IP-based transports in the ITC system or pertaining to a Cellular network, depending on context
CM	Connection Manager
CT or CTT	Closed Track [Test(ing)]
DCN	Document Control Number
DIST	Distribution
ELM	External Link Manager
EMP	Edge Message Protocol
EP or EPS	Endpoint [Simulator] or [Simulated] Endpoint, depending on context
ETE	End-to-End
FCC	Federal Communications Commission
GPS	Global Positioning System
GUI or UI	[Graphical] User Interface

Acronym	Description
HPQC	Hewlett Packard Quality Center
HRX	Host Radio Exchange
HW or H/W	Hardware
ID	Identification or Identifier, depending on context
IEEE	Institute of Electrical and Electronics Engineers
ILT	Integrated Lab Test[ing] or Lab Integration Test[ing]
IP	Internet Protocol
IS	Interchange Subsystem (part of ITCM)
ISMP	Interoperable Systems Management Protocol
ITC	Interoperable Train Control
ITCM	ITC Messaging Subsystem
ITCnet	ITCR's Air-Interface Protocol
ITCR	ITC 220 MHz Radio [Network] Subsystem
ITCSM	ITC Systems Management Subsystem
IWS	Integration Workstream
KES	Key Exchange Service
KPI	Key Performance Indicator
L	Locomotive Radio or Communication Site, depending on context
LSI	Locomotive System Integration
MCC	Meteorcomm LLC
Mgmt or MNG	Management
MPLS	Multiprotocol Label Switching
MSG	Messaging (same as ITCM)
n/a	Not Applicable
NAK	Not Acknowledge[d]
OS	Operating System
OTA	Over the Air

Acronym	Description
PER	Packet Error Rate
PTC	Positive Train Control
QoS	Quality of Service
R1.x	Release 1.x, where “x” is 0, 1, etc.
RAD	Radio (same as ITCR)
REQ	Request
RES	Response
RF	Radio Frequency
RHEL	Red Hat Enterprise Linux
RSSI	Received Signal Strength Indicator/Indication, depending on context
Rx	Receiver or Receive, depending on context
RWS	Radio Workstream
SBC	Single-Board Computer
SI or SIT	Systems Integration [Test(ing)]
SITB or TB	[Systems Integration] Test Bed
SMA	Systems Management Agent (this specific term has been shortened to “agent”)
SMS or SM	Systems Management [System] (same as ITCSM)
SOW	Statement of Work
SPx	Sprint x, where “x” is 0, 1, etc.
SQL	Structured Query Language
SVN	Subversion
SW or S/W	Software
TCP	Transmission Control Protocol
TE	Test Executive
TMC	Train Management Computer
TNS	Transport Network Subsystem (part of ITCM)

Acronym	Description
TRNS	Transfer
TTCI	Transportation Technology Center, Inc.
TTL	Time-to-Live
Tx	Transmitter or Transmit, depending on context
VPN	Virtual Private Network
W	Wayside Radio or Communication Site, depending on context
Wi-Fi	One of the IP-based transports in the ITC system or pertaining to the WLAN communication system based on the IEEE 802.11 standard, depending on context
WIU	Wayside Interface Unit
WLAN	Wireless Local Area Network
WS or xxWS	Workstream – where “xxWS” notes the: • ‘IWS’ = Integration Workstream • ‘MWS’ = Messaging Workstream • ‘RWS’ = Radio Workstream • ‘SMWS’ = Systems Management Workstream

Table 2: Terms and definitions

Term	Definition
Closed Track Testing (CTT)	PTC communication system testing that to-date takes place on non-revenue track at the TTCI test facility.
Direct Validation	Direct validation of a PTC communication system requirement is achieved when the expected result of test case execution ties to a specific ITC requirement.

Term	Definition
Endpoint	Endpoints include WIU, TMC, and BO applications and are represented by EPSs in the MCC IWS.
Indirect Validation	Indirect validation of a PTC communication system requirement is achieved when the nature of a test is such that multiple elements of the system must be functioning properly to achieve a positive outcome.
Fixed Site	Generic term for a BO/Base radio or a Wayside radio/communications site.
Functional Testing	Testing to determine whether a specific product feature is operational or not. Functional tests are typically pass/fail.
Kit	Specific set of configuration data and/or software files that can be downloaded to assets within the PTC communication system.
Mobile (Site)	Generic term for a Locomotive radio/communications site.
Performance Testing	Testing to determine how well some aspect of a product or system behaves under a particular workload as compared to a defined set of metrics. Performance metrics are quantitative in nature. Some examples of performance metrics that apply during integration testing are message latency and throughput.
Product Release Cycle	Refers to the development and testing activities within a major release of the PTC Communication System (R1.x).
Reference Hardware (HW)	Refers to the reference architecture SBCs employed to host remote ITCM and ITCSM SW and in the case of CTT, EPS SW.
Remote (Site)	Generic term for a Locomotive or a Wayside radio/communications site.
(Test) Scenario	A defined setup of product or system usage that approximates actual operation of the product in the field. Scenarios can be used to develop test cases. Typically, multiple test cases are derived from each scenario.

Term	Definition
Subsystem	With reference to MCC's development of the PTC communication system and in the context of integration testing, there are three subsystems: • 220 MHz PTC Radios (ITCR) • Messaging software (ITCM) • Systems Management software (ITCSM)
Test Bed	Refers to the specific hardware and software needed to test multiple ITC subsystems in an integrated test platform.
Test Case	Describes a specific test to be performed within a specified test environment and using a specific test configuration. The expected results of an executed test case support the validation of one or more requirements. The mapping of test cases to requirements can be one-to-one or one-to-many.
Test Configuration	Specific arrangement of test bed resources to facilitate execution of a specific test suite.
Test Cycle	Generically, a set of test suites grouped together for execution. Specifically pertains to the execution of all IWS test cases defined for a specific test platform in a given product release cycle.
Test Environment	Generic term that refers to the nature of a test facility. Typically, a test environment includes one or more test beds. Environmental variables can include physical as well as operational parameters.
Test Platform	Refers to the nature of one or more test beds as they relate to either a lab setup or field setup (ILT or CTT, respectively).
Test Procedure (or Process)	Detailed description of how to execute a specific test case (typically, step-by-step).
Test Scripts	A set of commands or SW elements that facilitate the execution of test cases.

Term	Definition
Test Sprint	The execution of test cases pertaining to a specific permutation of subsystem SW versions.
Test Suite	A group of test cases executed to accomplish specific technical or business goals.
Transport	Any of the various communications paths that the PTC communications system's traffic can be routed across from endpoint-to-endpoint. For the purposes of this document and MCC's IWS work, these include: • 220 • Cell • Wi-Fi • Hardwired Ethernet
Use Case	PTC communication system operation and activities that have significance to the user. Use Cases help define Test Scenarios.
Validation	A test process whereby specific product features, functionality, or performance levels are verified.
Workstream	A development program within MCC's PTC communication system development project. There are four separate work streams as follows: • 220 MHz PTC Radio Development workstream (RWS) • Messaging SW Development workstream (MWS) • Systems Management SW Development workstream (SMWS) • System Integration Test workstream (IWS)

1.5 References

The number in brackets [n] may be used within this document to indicate the referenced source document per the list that follows.

[1] *Communication System Integration Test Strategy* document, MCC DCN TSP-PTC-00001093-A, v0.4, 10/27/2010

[2] *PTC Communication System Lab Integration Test Plan* document, MCC DCN TSP-PTC-00001112-B, v0.0, 12/23/2010

[3] *PTC Communication Systems Closed Track Test Plan* document, MCC DCN TSP-PTC-00001163-B, v0.1, 3/2/2011

[4] *Closed Track Test Plan Review* presentation, MCC DCN TSP-PTC-00001219-A, 2/17/2011

[5] *IST 1.1 Integration Test Plan* document, MCC DCN 00001819-A, v1.0, 2/10/2012

[6] *Release 1.2 Integration Test Plan* document, MCC DCN 00002127-A, v1.0, 6/15/2012

[7] *ITCR Closed Track Test 1 Test Report* document, MCC DCN 00001825-A, v1.0, 2/14/2012

[8] *ITCC Release 1.0 Test Report – Integration Test* document, MCC DCN 00001946-A, v0.1, 2/24/2012

[9] *ITCC Release 1.1 Test Report – Integration Test* document, MCC DCN 00002377-A, 1.0, 4/24/2012

2. System integration test approach

The MCC PTC communication system contains three primary subsystems: 220 MHz PTC Radios (ITCR), Messaging software (ITCM), and Systems Management software (ITCSM). Integration testing of the communication system refers to a process under which multiple subsystems are combined together and tested as an integrated system.

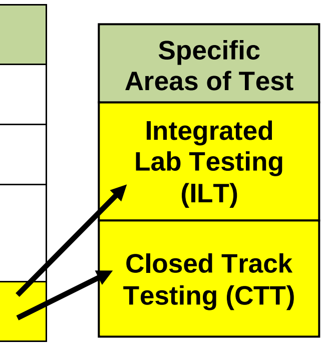
The PTC communication system development program consists of four primary workstreams. The first three workstreams are dedicated to the development of each subsystem and include the Radio Workstream (RWS), Messaging Workstream (MWS), and Systems Management Workstream (SMWS). The fourth, the Integration Workstream (IWS), brings the output products of the 3 subsystem workstreams together into a single integrated system for testing.

There are two environments that MCC's IWS uses for integration testing. The first is integrated lab testing (ILT), which occurs in a lab environment containing multiple test beds. The second is closed track testing (CTT), which occurs at a facility that allows the testing of OTA system operation with actual locomotives on a non-revenue track.

Table 3 shows the various PTC communication system development program's workstreams and their respective level of testing. IWS and its integration test activities (ILT and CTT) are the focus of this document and as such, are highlighted in Table 3.

Table 3: MCC testing universe

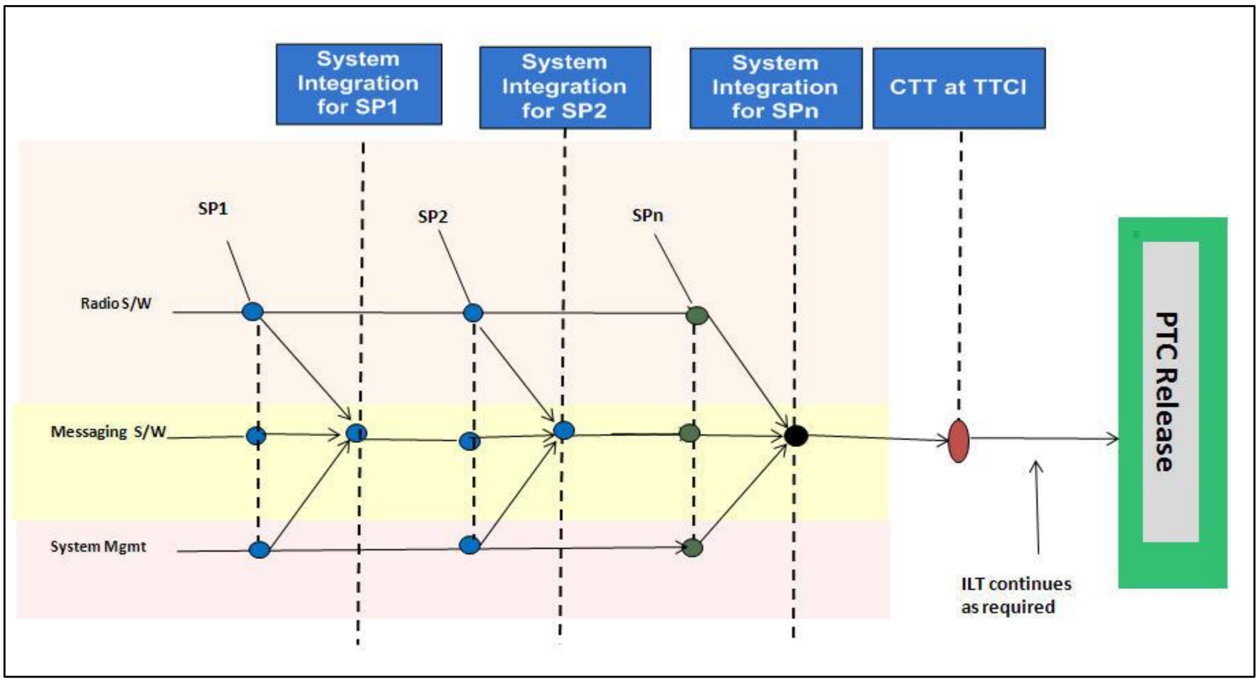
Testing Level	Workstream	Name
Subsystem level	RWS	Radio (ITCR)
	MWS	Messaging (ITCM)
	SMWS	Systems Management (ITCSM)
System level	IWS	System Integration Test



Incremental functionality of the PTC communication system is based on product releases. For example, Release 1.0 (R1.0) was the initial release of the PTC communication system at the "green line" in October 2011. Subsequent major releases are R1.1 in May 2012, R1.2 in August 2012 and future releases such as R1.3, R1.4, etc. Per release, new requirements are levied and as a result new functionality is implemented and tested. In general, functionality added in the most recent release builds on the functionality of previous releases.

Figure 1 depicts the integration of the subsystem workstreams as it pertains to a specific product release. The three work streams (RWS, MWS, and SMWS) work independently doing subsystem development and testing. The subsystem products (typically software and associated tools) are periodically released after "sprints" of development. After these releases, the subsystem products are integrated into a system and that integrated system is tested as a whole in the IWS.

Figure 1: Integration of MCC's subsystem workstreams



During each major release, not only is the integrated system repeatedly tested in the lab, but it is tested in a controlled field environment, too. These are the integrated lab testing (ILT) and closed track testing (CTT) portions of MCC's integration test activities, respectively. As shown in Figure 1 the PTC communication system typically undergoes ILT before CTT. After several minor releases and successful rounds of ILT and CTT where intended functionality and defect fixes are verified, the system is officially ready for major release to the customer railroads.

2.1 System integration test objectives

The primary objectives for system integration testing are as follows:

- Ensure that the PTC subsystems (ITCR, ITCM, and ITCSM) work together and as designed in an integrated communication system
- Within the context of lab (ILT) and field environments (CTT), validate that the PTC communication system meets system performance levels, features, and functionality as agreed upon per the requirements of each product release
- During lab and closed track testing, provide results and triage defects to facilitate product updates, fixes, and improvements

Secondary objectives for system integration testing are as follows and largely reinforce the primary objectives noted above:

- Resolve usability, logistical, and functional issues that may negatively impact the next phase of testing, next product release, or real-world deployment and subsequent operation
- To provide both lab-based and field-based test beds for continued PTC communication system testing

Note that both Sections 1.2 and 1.3 (Assumptions and Scope) help reinforce the objectives of MCC's IWS and its integrated system test activities.

2.2 Successive validation

A key component of the MCC PTC test approach is the concept of successive validation. This is a process whereby the verification of the functionality and performance of design elements and subsystems is performed before moving to the next level of integration. As such, system-level integration follows a staged approach where basic integrated functionality is validated before incorporating more system elements. Examples of this for the system integration test process include the following:

- Validation of peer-to-peer communication between each radio type before incorporating messaging SW
- Validation of messaging between a base and a locomotive before turning up a wayside

- Obtain a system functionality baseline with a smaller network (for example, 3 radios or communication sites) before going to a larger number of network elements (such as 10 radios or communication sites)

Once initial integration between components is validated, additional complexities are incorporated into the test lifecycle using this controlled staging process.

2.3 Entrance and exit criteria

Product performance is continually evaluated as it progresses through the MCC test process. Between phases in the PTC product development program, the product is evaluated against a set of criteria before proceeding into the next phase and/or out of the current phase. A couple of examples of where this occurs in the MCC test process include the following:

- The point at which each subsystem is released to integration
- The point within integration when a specific release is passing from ILT to CTT

Figure 2: High-level flow through the system integration test process

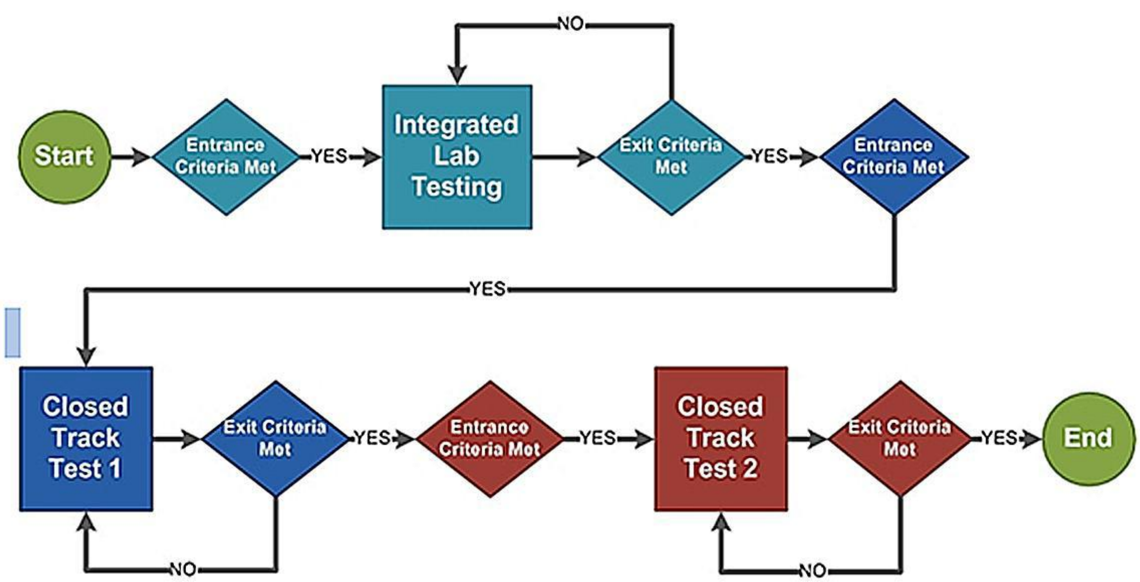


Figure 2 illustrates this process where successive phases in integration testing have discreet entrance and exit criteria between each phase.

(Note the following with regard to Figure 2: "Start" represents the point at which the product is released from the subsystem(s); the concept of two stages of CTT is dependent on the need and resourcing pertaining to a specific release and as such, does not apply to all product release cycles.)

2.3.1.1 Entrance criteria

The criteria for entering system integration test are based upon the subsystems meeting a minimally acceptable level of performance. Basic criteria are that each subsystem has executed its particular test plan and has validated the performance of the subsystem as a whole to an appropriate level.

Likewise, the criteria for moving from one phase of integration testing to another (for example, ILT to CTT) are based upon the test results of the initial phase meeting a minimally acceptable level of performance. Additionally, the entrance criteria of each integration phase include testing preparation as well as verifying subsystem functionality and performance levels.

Examples of entrance criteria are shown in the following tables where Table 4 shows an overview of entrance criteria through integration testing phases and Table 5 shows more detail regarding specific subsystem validation levels. Note that the information in these tables is for discussion purposes only. Complete entrance criteria are defined in reference documents [2], [3], and [5].

Table 4: Entrance criteria overview for integration's test phases

Dependency	Entrance Criteria
HW Equipment Availability	• MCC's PTC Radios • Servers/SBCs to host needed remote SW • Servers for BO setup and test control • IP equipment such as modems and Ethernet switches
Test Control	• Available test tools for connectivity, control, and data capture, such as: o Test Executive (TE) o Endpoint Simulator (EPS) o Radio Control Application

Dependency	Entrance Criteria
Test Environment	• Test Bed(s) built up • Test scripts created and validated • IP topology in place and validated ○ IP connectivity to all components (for example, PTC radios, ITCM computers, ITCSM components) in all assets' comm-sites from a central location (not via 220 MHz) ○ As alternative ITC message paths (transports) as well as for test control
Test Environment (CTT specific)	• Necessary test facility infrastructure is in place and ready for use such as all antennae and associated cabling for GPS, 220 MHz, Cell, and Wi-Fi • 220 MHz spectrum usage verified
Test Methodology	• Test Plan completed and approved • Previous test phase completed with its exit criteria met (including any waivers as applicable)
Subsystem Functionality (high level)	• ITCR, ITCM, ITCSM and their associated protocols ○ Applicable versions' SW testing done ○ No critical defects[i] and high confidence ○ Subsystem KPIs exceed 75% of required levels • Refer to Table 5 for more detail on Subsystem Functionality

i. Note that this is in the context of what subsystem functionality is available, which could be rudimentary to advanced depending on the specific product release

Table 5: Subsystem entrance criteria for integration

Dependency	Entrance Criteria
Product Quality	• No Priority 1 or 2 defects remain in open status unless by a mutual agreement waiver • Total defect count and number of open defects at each severity level is noted

Dependency	Entrance Criteria

Test Coverage	• All requirements are traced to test cases
	• Baseline requirements test coverage = 100%
	• Derived requirements test coverage = 100%
	• Subsystem testing has been completed
	  ○ All subsystem development team test cases have been executed
	  ○ All subsystem test team test cases have been executed
	• Test results have been summarized and made available
	• Regression testing of previous release requirements = 100%

Functionality	• Required functionality = release is code complete
Tools	• For installation and configuration • Deployment tools created, tested, and validated • Installation process and documentation have been verified
Performance	• ITCM KPIs have been baselined in a VM environment and real environment with SBC remote messaging servers • ITCnet and radio KPIs have been baselined • Basic radio operation and performance includes adequate Rx sensitivity and Tx power levels • Testing of SW components needed to support OTA asset updates and configuration

2.3.1.2 *Exit criteria*

Applying exit criteria to the PTC communication system or subsystem at the end of a test phase works similarly to that of applying entrance criteria at the beginning of a test phase. Both evaluate the results of what had just been done with an eye to what that means with regard to system functionality and performance going forward. Although minute, the primary difference would be that exit criteria tend to concentrate on the former (evaluating what was just done) whereas entrance criteria focus on the latter (what that means going forward).

As such, exit criteria include defined levels of functionality and performance. Examples of exit criteria are shown in Table 6. Note that the information in Table 6 is for discussion purposes only. Complete exit criteria are defined in reference documents [2], [3], and [5].

Table 6: Exit criteria overview for integration's test phases

Dependency	Exit Criteria
Test	• All priority 1 and 2 test cases have been executed
Test Coverage	• Priority 1 test case pass rate = 100%[i] • Priority 2 test case pass rate = 85%[i]
Product Quality	• All critical and major (priority 1 and 2) defects have been resolved
KPIs	• Applicable system-level KPIs meet or exceed required levels to a 75%[i] confidence level (more specifically, KPI measurements will likely meet required levels 75%[i] of the time)
Reporting	• Test results have been documented and archived

i. Note that these values are specific to particular releases or phases within a release and are listed here for discussion purposes only. As the PTC communication system matures through successive test stages and release cycles, these values are typically increased.

A status check of test progress against exit criteria occurs on a periodic basis, happening more frequently as a test stage nears completion. Ultimately, it is expected that the status of the exit criteria will be more formally reviewed by a cross-functional team that includes test, product development, and engineering services.

2.3.1.3 *Criteria evaluation*

Entrance and exit criteria are not intended to be a hard gate before proceeding from the current or to the next stage in the MCC test process. Rather, this criterion defines a level of functionality and performance needed to allow the next test phase to proceed with a minimum of issues.

Additionally, the purpose of having entrance and exit criteria is to:

- Provide an internal MCC milestone for an in-depth cross-functional status review of the PTC communications system and its subsystems
- Communicate key dependencies to all stakeholders by ensuring that the development teams, test teams, MCC management, and other stakeholders have sufficient information to evaluate risks

- Identify critical logistical elements such as:
  - Providing guidance on completion targets to the various workstreams
  - Allowing anticipated levels of available functionality to be estimated for test planning

Whether it pertains to ILT, CTT, or MCC's IWS testing as a whole, in the event that all entrance or exit criteria are not met by the time a test phase is set to begin or end (respectively), exceptions may be made and are reviewed on a case-by-case basis. The guiding principal is to remove barriers to moving the product to the next phase in the PTC communications system program without compromising product quality. It is understood that issues and defects associated with waived criteria are investigated and addressed by the corresponding workstream(s) in parallel with ongoing MCC testing and/or railroad activities.

2.4 Integration test phases

The concept of successive validation (introduced in Section 2.2) is exemplified throughout the organization of the whole PTC system project as all of its testing is split into phases and stages. For example, MCC's primary focus is the PTC communications system, which includes testing from product development through subsystem and integrated system stages into support of the railroads as the addition of real endpoints and full deployment activities are started up. A high-level staged approach to the overall PTC system testing is noted as follows. The first primary bullet is specific to the PTC communications system where its last sub-bullet is specific to MCC's integration activities:

- PTC communications system testing (focus of MCC).
  - Individual component test and characterization (focus of MWS, RWS, and SMWS).
  - Subsystem test (focus of MWS, RWS, and SMWS): Components are successively added until the full subsystem is created and comprehensively tested.
  - Integrated system test (focus of IWS): Subsystems are integrated into a complete communication system for system-level specific testing. IWS activities occur in the following environments.
    - Lab (ILT)
    - Field (CTT)

- Full PTC system test (focus of railroads with MCC support): The PTC communications system is outfitted with real endpoints and ultimately fully deployed.

Per each phase noted above, as initial integration between the last phase's elements is validated, additional complexities are incorporated in the test lifecycle using a controlled staging process. For IWS activities, ILT and CTT are successive phases and have discreet entrance and exit criteria between each phase (as detailed in Section 2.3).

2.4.1 Integrated Lab Testing and Closed Track Testing overviews

Integrated Lab Testing (ILT) focuses on system-level tests of the PTC communication system in a lab environment. The testing is directed towards characterization of system functionality and evaluation of system performance. Particular emphasis is placed on test scenarios that involve cross-subsystem interactions (for example, validating proper functionality of a protocol that connects two subsystems such as HRX). In support of this, the guidelines and principles detailed in Section 1.3 (Scope) have been applied.

Though some of the Closed Track Testing (CTT) scenarios are similar to those in ILT, CTT differs from ILT in two important ways:

1. Radios communicate over the air rather than through a cabled RF network
2. The system is operated in a real track environment using real locomotives traveling at varying speeds

These characteristics allow a number of tests to be conducted that are not as well performed in a lab environment. As such, CTT focuses on mobility, direct peer-to-peer messaging while moving at speed, system robustness in the presence of real intermittent OTA connections, ETE messaging in a field environment, and OTA asset updates and management.

2.4.1.1 Primary objectives

The primary objectives of IWS's ILT and CTT are very similar to and are in direct support of the overall IWS objectives noted in Section 2.1. In fact, all ILT and CTT activities are planned and executed with the IWS objectives in Section 2.1 and the following primary objectives in mind:

- ILT: To validate that the PTC communication system, with all subsystems integrated together, functions in accordance with system requirements within a controlled lab environment.
- CTT: To validate that the PTC communication system, with all subsystems integrated together, functions in accordance with system requirements within a controlled field environment.

2.4.1.2 Successive validation in ILT and CTT

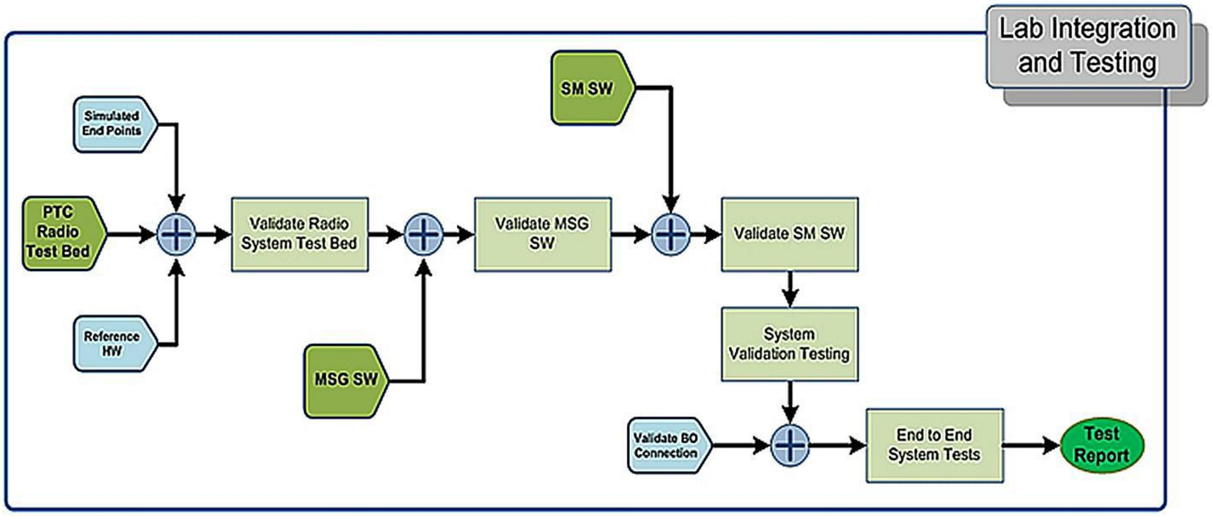
As noted in previous sections, MCC's testing of the PTC communication system is based on a process of successive validation whereby system elements are verified prior to proceeding to the next level of integration with more added complexity. As such, IWS testing is not intended to revalidate features and functionality already well tested at the subsystem level. Rather, ILT and CTT activities will evaluate system performance of the integrated PTC communication system in realistic but controlled environments using system-level test scenarios.

Given the subsystems' version level in a particular release cycle, the PTC communication system is tested in ILT prior to CTT. ILT is performed using test beds that simulate as much as possible the connectivity and data traffic environment that the PTC communication system will encounter in the field. In support of this, ILT is performed using simulated endpoints (EPSs that simulate WIU, TMC, and BO applications).

Figure 3 represents the successive validation process for ILT. Each "plus" symbol on the diagram represents a stage where integration is validated prior to incorporating additional complexities. For example, the first plus in the process is to validate that the PTC radio test bed, EPSs, and reference HW single-board computers are functioning as a complete radio system test bed. Once the test bed has been validated, the messaging SW is added to the environment and messaging functionality is validated. Systems Management capabilities are then integrated into the system and validated as well. System-level test cases are performed to validate the entire system at this stage. After that validation is completed, a BO connection is added and ETE system tests are performed.

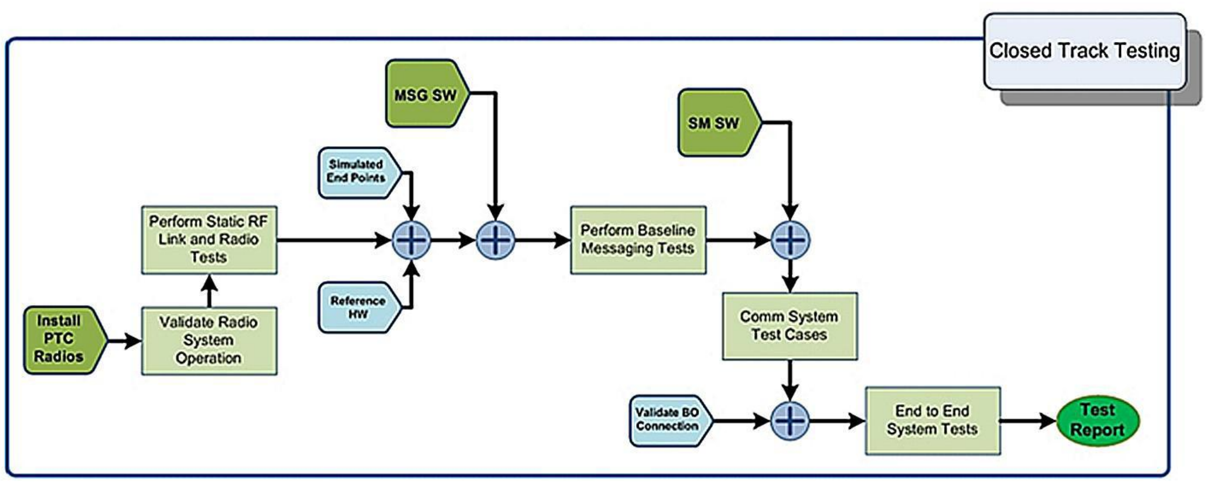
(Note that Figure 3 is for illustration purposes and is not intended to imply any quantitative timeline information. Also, the actual process followed for ILT may vary.)

Figure 3: ILT process of successive validation



After validation in ILT, testing moves out to CT, which is a test bed in a controlled field environment that includes actual railroad and terrestrial communication infrastructure such as real locomotives, actual track (non-revenue), antennae, wayside, and base sites. CTT expands on ILT in the functional areas where OTA testing adds value in supporting the end-result of a properly performing PTC communication system.

Figure 4: CTT process of successive validation



Similar to ILT, CTT will be performed incrementally by adding complexity factors and validating each step in the testing cycle. Figure 4 represents the successive validation process for CTT.

(Note that Figure 4 is a process diagram for illustration purposes, and is not intended to imply any quantitative timeline information. Also, the actual process followed for CTT may vary.)

2.4.2 Integration test's preparation stages and activities

Implicit in the process of successive validation is the fact that initial phases and stages within IWS focus on evaluating the PTC communication system performance and limitations, while at the same time refining the test environment, data collection, and analysis capability. The results of which provide feedback for improvements and adjustments in test control, data collection capability, and ultimately PTC communication system performance. These improvements and adjustments are applied during subsequent IWS phases and stages. With that said, both ILT and CTT are conceptually divided into multiple stages where completion of each stage helps ensure the next stage's testing proceeds with minimal issues.

2.4.2.1 Pretesting and shakedown testing

The initial stage of either ILT or CTT, typically referred to as pretest or "shakedown", includes activities such as validation of the test infrastructure and basic evaluation of early subsystem releases. The primary goal of this stage is to produce a reliable test bed. As such, this stage focuses on the execution of test cases needed to validate the test bed and resolve any associated issues.

In the R1.0 product release cycle, an appreciable part of this stage included the initial bring-up and commissioning of the associated test platform (for example, validating test bed infrastructure, basic connectivity, and test-control operation). A number of these activities were respectively completed in advance of the start of the R1.0 ILT and CTT phases. Some key elements of this preparation included the following where some are specific to the field setup at CT as opposed to a lab environment:

- Installation of PTC communication system equipment
- Installation and operational validation of all test control and monitoring systems
  - Establishment of the remote connection between the CTT facility and MCC in Renton, Washington

- Obtaining the rights for OTA transmission on the frequencies in the 220 MHz spectrum at the CTT facility
- Basic validation and analysis of the physical connectivity between PTC communication sites

Activities around equipment installation and commissioning are typically a one-time need, unless of course, the equipment or component is in need of replacement or a major update. As such, activities surrounding validating test bed infrastructure still take place at the beginning of a test phase. However, as can be deduced, its emphasis tends to decrease as the testing of subsequent product release cycles continues.

Alternatively, evaluation of early subsystem releases is continually emphasized during shakedown testing. In fact, the validation of subsystem functionality is so core to reducing risk of system-level test issues that its emphasis has increased through subsequent product release cycles since R1.0. This increasing emphasis is discussed in the following sections. Table 7 lists the preparation activities requiring shakedown testing to validate either the ILT or CTT test platforms.

Table 7: Preparation, pretest, and shakedown test steps

Step	Description	Notes
1	Get all agreements into place	• This includes all needed vendor and regulatory agreements such as spectrum use (FCC and PTC220, LLC), CTT facility use, and Cellular use

2	Assemble/Install test bed(s) infrastructure	• This takes place well in advance of installation of the PTC communications system
		• For ILT, this entails test rack assembly and basic cabling into MCC's networks and power
		• For CTT, this entails:
		  o Ensuring that site infrastructure is in place such as backhaul connectivity and power as well as antennae and cabling installation on the towers and poles
		  o Establishing a VPN remote connection from MCC to CTT facility

Step	Description	• Notes
3	Validate basic IP	• Focus is connections between test control and various communication sites such as the BO, base sites, and the remote sites • For CTT, focus is on the VPN connection between MCC's remote test control and various CTT facility sites

4	Install and configure PTC communications system and test-control equipment in test bed(s)	• Focus is on HW with SW as needed based on the subsequent validation steps. This entails:
		  ○ Test control and BO servers
		  ○ 220 MHz Radios
		  ○ Remote site SBCs to host SW such as ITCM, SMS, and EPS SW
		  ○ Additional IP equipment such as remote site Cell and Wi-Fi modems

5a	Install, configure, and validate test-control SW tools and applications	• Validate operation between applications as well as the relevant PTC communication system components and sites as applicable in the subsequent steps. This entails:
		  ○ TE
		  ○ EPS SW
		  ○ Radio control and monitoring tools
		  ○ Data logging tools
		  ○ Infrastructural control and monitoring tools

5b	Validate test-control connectivity	• PTC communications site to site especially via Wi-Fi and/or Cell back to test-control sites • App to app as well as basic IP
6	Validate 220 MHz radio network operation	• Validate proper radio functionality based on wanted configuration • ITCnet-specific messaging between radios
7	Load ITCM SW onto site-specific computers	• BO = Server • Remote = Reference HW SBC

Step	Description	Notes
8	Integrate ITCM Server or SBC into communication site	• Validate that the radio network and ITCM are still operating correctly alone and then together and then along with the EPS and TE • Validate BO ELM registration
9	Validate message transmission via the PTC radio network	• Messaging between communication sites over radio from ITCM host to host • Validate along with EPS and TE
10	Validate message transmission via IP paths	• Same as previous step but with: ○ Cell ○ Wi-Fi ○ Hardwired (as applicable)
11	Load SMS onto site-specific computers	• Reference Agent = SBC • Real Agent = Radio itself or ITCM SBC as applicable • BO = EPS Server
12	Validate connectivity between all SMS components	• ISMP connectivity • BO's ITCSM gateway to remote sites' various agents, which are component dependent (such as radio or ITCM SBC)
13	Install, configure, and validate additional BOs	• If applicable and as needed • Remote and/or foreign BO functionality

2.4.2.2 *Integrating subsystems versus system integrated test*

As noted in the previous section, due to its effect on subsequent system-level testing, emphasis on validating subsystem functionality continues to have an increasing emphasis in IWS test activities. This increased emphasis has created an intermediate stage in ILT, which for the purpose of this document, is referred to as subsystem integration or “integrating subsystems”. Much of this intermediate stage is more relevant to ILT as opposed to CTT since a lab environment offers a more comprehensive functional check on each subsystem with more ease, including fewer logistical problems and an easier environment to diagnose defects at less cost.

Subsystem integration is a process that brings together multiple components and subsystems through a series of defined steps to reach a specified objective. The objective of subsystem integration is to successfully integrate together versions of ITCM, ITCR, and ITCSM into an operational PTC communication system. This stage is completed when the functionality and stability of the system has reached a sufficient level of quality to allow meaningful system-level testing to begin.

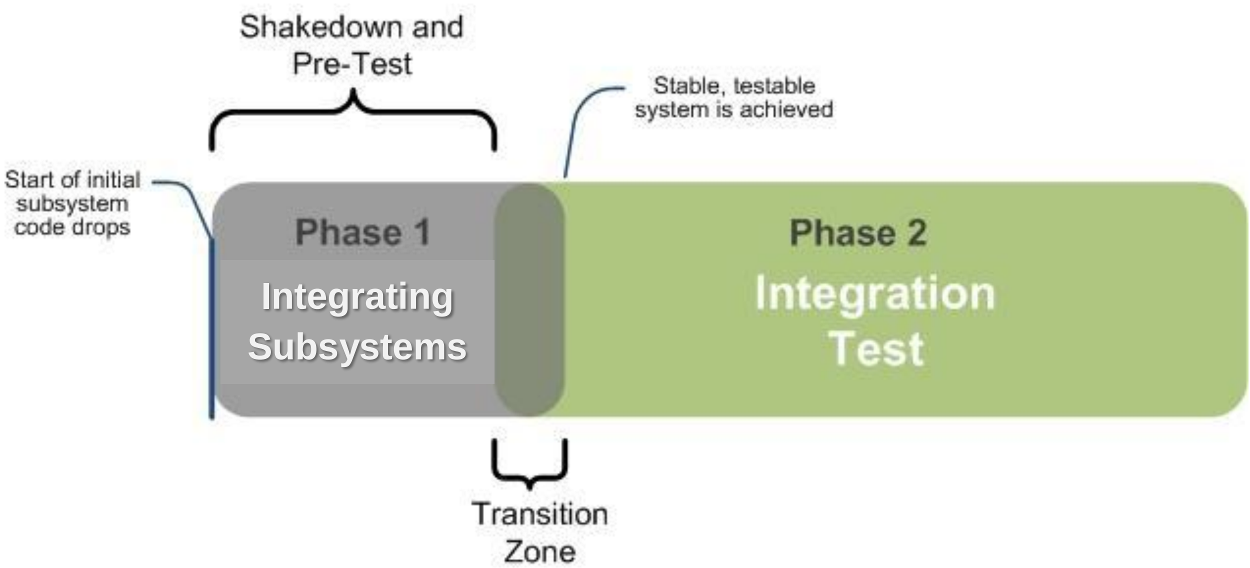
Integrating subsystem testing differs from full system-level integration test in that the former's goal is to *produce* a stable PTC communication system whereas the latter *starts* with a stable PTC communication system. As such, testing for integrating subsystems focuses on subsystem evaluation, subsystem-pair testing, and is directly affected by the IWS subsystem code-intake process – all of which are further discussed in the following sections. Conversely, full system-level integration test executes test cases that primarily validate ETE system functionality.

Following are some operating principles which need to be considered when integrating subsystems:

- Primary goal: An operational integrated system of sufficient quality to allow meaningful system-level testing to begin.
- Primary focus: To evaluate core functionality, thereby deemphasizing KPIs and system performance testing.
- Minimize deep-dive troubleshooting except where necessary to drive towards an operational system.
- Run tests that provide opportunities to identify critical defects early and target test cases at specific functional areas and features that exercise essential cross-subsystem interactions.
- Begin working with subsystem releases as soon as possible.
- Avoid transitioning out of subsystem integration and into full system integration testing too soon. Given the sophistication of the IWS test tools, testing of a stable system is highly efficient. As such, there is high value in maximizing system stability and quality prior to the start of full system integration testing.

Figure 5 presents the overall concept detailed above where integrating subsystems occurs first, followed by actual system-level test of the integrated PTC communications system. Figure 5 also indicates that the completion of subsystem integration and the start of integrated system test is not a clearly defined point, but rather a transition where these two stages overlap.

Figure 5: Transitioning from integrating subsystems to integrated system test



2.4.2.3 Subsystem testing in IWS

A key focus of MCC's IWS activities is to validate core functionality and identify as many critical product defects as possible. Of particular interest are those defects that block testing of core functions. Though it is desirable to begin evaluating all three subsystems integrated together as early as possible, prior to that activity or in parallel with it, evaluation of subsystem releases as separate elements and in pair testing can provide significant benefits:

- Isolates tests to specific areas
- Simplifies the test environment, making issues easier to diagnose
- Evaluates an unknown subsystem under known conditions
- Evaluates an unknown subsystem in operation with a known subsystem

Additionally, pair testing is a valuable process for vetting test procedures and enhancing test control as supported by the following examples:

- Verification of test script execution
- Dry runs of various test cases
- Validation of tools and simulators

At each new code drop an evaluation is made to determine what level of subsystem testing and pair testing is needed.

(a) *Single-subsystem evaluation*

Since ITCSM requires ITCM to function, evaluation of single subsystems separately applies to only ITCR and ITCM. The following lists some examples of IWS's single-subsystem evaluation:

- ITCM-only tests:
  - Validate proper installation
  - Test messaging across IP-only paths of Cell, Wi-Fi, and wired if applicable (not radio)
  - Run tests that stress functionality of ITCM's specific components which play a large part at system-level (for example, CM, TNS, IS, AG, etc.)
- ITCR-only tests:
  - Validate that radio configuration took via radio-specific test tools such as XTermW or the PTC Radio GUI (for example, channels, master base, whether the radio's hearing ITCnet overhead traffic from who it should be, etc.)

(b) *Subsystem pair testing*

As conceptually shown in Figure 5 full system integration testing follows integrating subsystems testing. Though system integration test focuses on ETE tests that include all three subsystems, some pair testing activity is expected to continue in parallel.

Since ITCSM requires ITCM to function, subsystem pair testing applies to only ITCM-ITCR and ITCM-ITCSM. Some subsystem pair testing is of more value than other pair testing. Prior to full system-level integration testing the focus of pair testing is as follows:

- After basic validation and confidence testing, prioritize pair tests that evaluate cross-subsystem interactions (that is, testing across the integration "touch points").
- Based on understanding of significant product architecture or code changes from the previous product release cycle, run pair tests that specifically evaluate these areas. In some cases a high level of

regression testing is expected. (The concept of regression testing is discussed in more detail in Section 3.5.2.)

After integrating subsystem testing is complete and full system-level integration testing is underway, pair testing is used as follows:

- Confidence testing of new code drops
- Trouble shooting issues seen with full system integration testing
- Investigating cross-subsystem interaction and issues in a simpler environment than that of a fully integrated system

The following examples of pair tests are run prior to and in parallel with testing of a fully integrated PTC communication system:

- ITCM-ITCR pair tests:
  - Basic Validation and non-ITCSM-related confidence testing
  - Radio-focused testing (for example, messaging over the 220 transport)
  - Regression testing of the prior release cycle's ITCM-ITCR test cases
  - Testing of the current release cycle's new ITCM-ITCR-specific functionality
- ITCM-ITCSM pair tests:
  - Perform basic asset-related tests for the remotes that focus on ITCSM gateway interactions with both the remotes' reference agent and ITCM agent
  - Proceed to confidence testing
  - Regression testing of the prior release cycle's ITCM-ITCSM test cases
  - Testing of the current release cycle's new ITCM-ITCSM-specific functionality
- Additional pair testing that may provide value and can be opportunistically addressed includes testing the above subsystem pairs in various backwards-compatibility permutations (for example, R1.0 ITCM and R1.1 ITCSM, R1.1 ITCM and R1.0 ITCR, and so on). (The concept of backwards compatibility testing is discussed in more detail in Section 3.5.3.)

2.4.2.4 Subsystem code-intake process for IWS

During integration testing throughout each product release cycle, there are periodic code drops from the subsystem development teams. Figure 6 shows this from a holistic product release cycle as it pertains to IWS activities.

(Note the following with regard to Figure 6: Although shown for the R1.1 and R1.0 product releases, its concept is applicable to IWS's intake of subsystem code drop for all product releases. It is for illustration purposes and is not intended to imply any quantitative timeline information.)

Figure 6: IWS's subsystem code-intake during a product release cycle

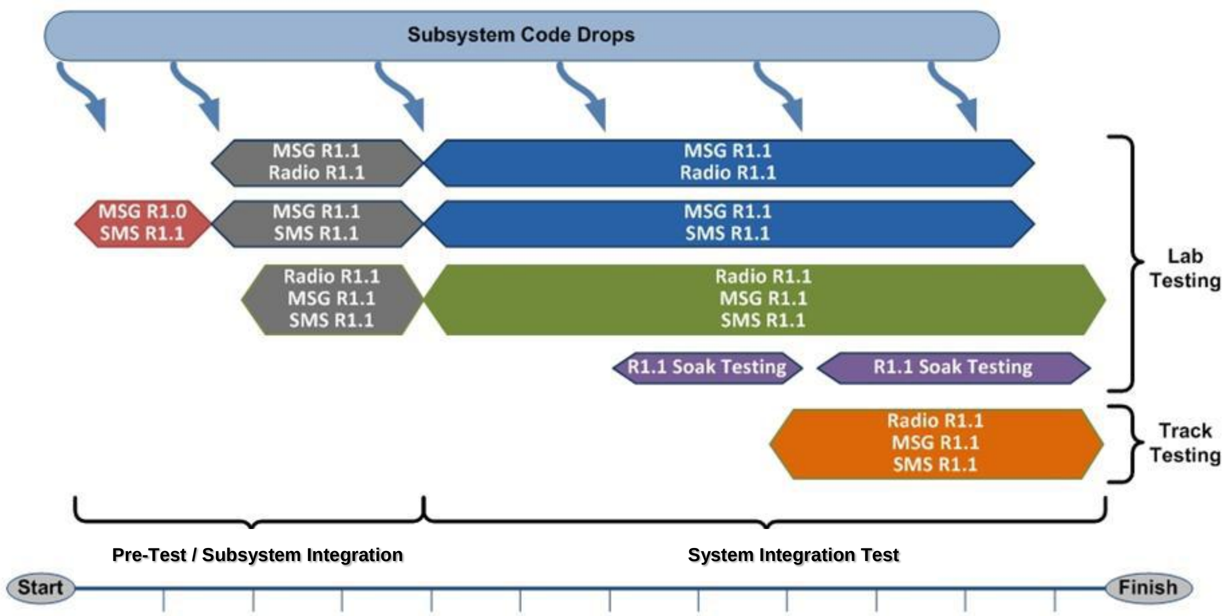


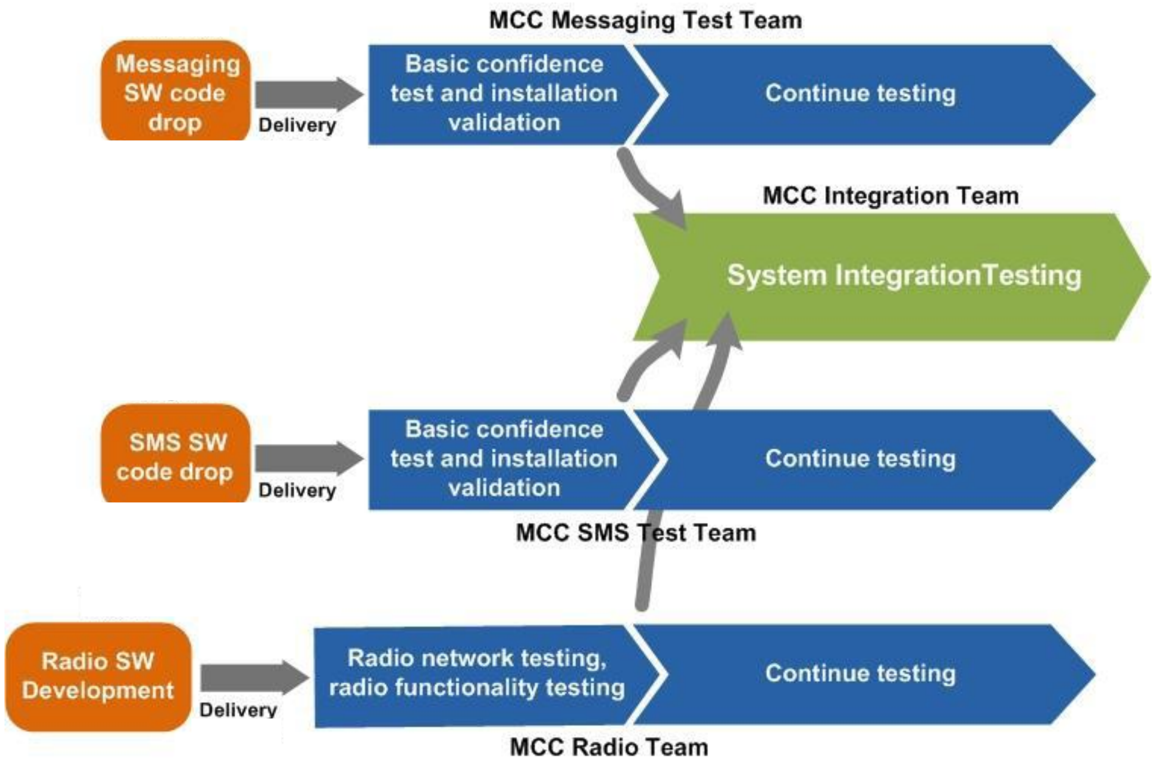
Figure 7 shows the basic flow for the code drop process for all subsystems. Following is a summary description of the process:

- ITCM and ITCSM code drops are delivered to MCC from the corresponding subsystem development team.
- The MCC ITCM and ITCSM subsystem test teams first perform basic validation of the code drop. This typically includes validating the SW installs properly and basic functionality checks. If no significant issues are identified, the code drop moves on to the integration test team and the subsystem test teams continue with the remainder of their test plans in parallel.
- The MCC radio development team generates the ITCR code drop and also tests it prior to its delivery to the integration test team. If no significant issues are identified, the code drop is given to the

integration test team and the radio test team continues with the remainder of their test plans in parallel.

- Depending on the level of urgency, the specific defects that are fixed in a particular code drop, and the anticipated level of SW stability, one or more of the subsystem code drops may be delivered to the integration test team prior to completion of basic validation.

Figure 7: Code-drop process into system integration test



Following is the basic process for IWS when it takes receipt of new code drop:

1. The new SW is installed on a single test bed.
2. Basic functionality is validated. This includes tests directed at specific subsystem functionality as well as tests that evaluate operation at a system level. The intent is that these tests do not duplicate the basic validation testing performed by the subsystem test teams.
3. After confidence in the new code drop is sufficient, the new SW is propagated out to more integration test beds.

If code drops from two or more subsystems are delivered to integration at the same time, then each subsystem code drop can be validated on a different test bed. Since there are a limited number of test beds, step #3 above may require careful coordination as each code drop is propagated out to other test beds.

2.4.2.5 Software (SW) permutations

Within a particular product release cycle (R1.x), a subsystem code drop indicates a minor version change in that subsystem's SW. For example, successive ITCM code drops within R1.1 changed the ITCM SW version being used in IWS from 1.1.0 to 1.1.1 to 1.1.2 and so on, all of which are unique minor versions of R1.1 ITCM SW.

New subsystem code signifies functionality has been added or modified, sometimes in the form of fixes to defects discovered in a previous SW version. As such, a new SW drop of at least one subsystem changes the subsystem SW permutation in integration test and fundamentally indicates an alteration to the whole PTC communication system's operation and capabilities. IWS testing is conducted within the context of subsystem SW permutations.

To minimize the impact on integration testing efficiency, the IWS plans on the subsystem WSs providing SW drops at the same time. This maximizes the amount of integrated testing that can be conducted on a specific SW permutation. Typically, the subsystem SW drop dates into IWS are planned to occur at some regular cadence throughout the product release cycle. In between these drop dates; ILT conducts a specific "test sprint". There are multiple ILT test sprints in a given product release cycle. Typically, in practice, CTT occurs with one subsystem permutation (within a single test sprint) near the end of the product release cycle.

Two goals of each test sprint are to discover defects for triage while also validating PTC system functionality and performance. Although there is overlap between these goals (as they often go hand-in-hand), the former effort is generally elevated in priority in the earlier sprints of the product release cycle, whereas the latter effort is elevated in priority as the product release cycle nears its end.

In an effort to reduce risk of defect discovery after release to the railroads, a collective goal of all ILT test sprints is complete test coverage. This means that an entire ILT test cycle has been completed before release thereby implying all ILT test cases have been conducted. As such, outside of basic functionality testing (as noted in #2 of the previous section), testing in a new sprint typically starts off in the ILT test cycle where the last sprint ended. This enables complete test coverage sooner in the product release cycle, which means defects can also be discovered, triaged, and resolved sooner.

The following is an example to further clarify this concept. The specific quantities are illustrative only:

- Assume the ILT test cycle for R1.4 includes 1000 test cases, the first 100 of which are basic functionality test cases.
- Assume R1.4's ILT 'Test Sprint 1' has executed the first 400 test cases in the ILT test cycle at a time when new SW drops for all 3 subsystems occur.
- Upon receipt, IWS validates the new SW drops, which includes rerunning the 100 basic functionality test cases as the start of R1.4's ILT 'Test Sprint 2'.
- After the 100 basic functionality test cases are done, 'Test Sprint 2' picks up at test case 401.
- If 'Test Sprint 2' conducts test cases 401 through 1000 and time is still left before the next SW drop(s), testing in 'Test Sprint 2' would proceed to test case 101 and run up through test case 400 as possible.
- At the next new SW drop(s), the process starts again as 'Test Sprint 3'.

2.4.3 Scope of systems integration testing

The primary stage in either ILT or CTT is actual test case execution of the fully integrated PTC communications system – the primary duty of MCC's IWS. This stage follows the preparation stages that were detailed in the previous sections. With a stable PTC communication system in place, testing that focuses on system level evaluation and characterization now begins in earnest. This is accomplished by executing test cases that primarily consist of ETE system tests.

Table 8 shows the specific test scenarios that are the primary focus of MCC's integration activities.

Table 8: Summary of integration test scenarios

Test Scenario	Description/Conditions
Validation of Previous Product Release Cycle	• Prior defect checks which include validating fixes of all IWS-found defects since the last cycle's final release • Regression testing including all system-level functionality in the previous product cycle as well as applicable subsystem pair testing

Test Scenario	Description/Conditions
Basic Functionality	• 220/ITCnet connectivity and radio network operation • IP connectivity such as Cell and Wi-Fi paths • ELM registration with and without ITCnet • Broad range of tests to validate various system elements and basic system operation including aspects of many of this table's following scenarios, which are tested in depth as noted once testing proceeds beyond basic functionality.
Messaging	• Outbound/Inbound • Broadcast and unicast messages • QoS metrics such as network preference, message class, and flooding • Rerouting and route dampening • Transport control and cost metrics • Message delivery during transport fail-over • Message priority and expiration/TTL effects
Fragmentation	• Long message delivery tests over various paths and in various directions • Large SMS kits sent to remotes
Interoperability	• Federation including Tier-1/Tier-2 • Message delivery involving home and foreign assets over various transports • Messages direct to home or foreign BO • Messages to foreign BO via home or other foreign BO
Peer-to-Peer (PTC)	• Special handling messages (short and long broadcasts) • Wayside beacon messages • WIUStatus messages (requests and responses) • Beacon On messages • Direct and virtual wayside paths • With or without Base radio coverage

Test Scenario	Description/Conditions
Mobility	• Locomotive base selection • Message delivery during locomotive handoffs • With same or different ELM(s) and BO(s) • Messaging while going into and out of base coverage • High locomotive speeds up to 107MPH • Effects of RSSI, position, and connectivity failures • Effects on KPIs such as latency
RF Links	• Static and Mobile • 220 MHz RF link performance including BER and PER as a function of locomotive speed • RF coverage along the test track and radio-to-radio communication (L<>W, B<>L, B<>W) from geographically distributed communication sites
Communication Failures	• Link and radio failure scenarios with and without alternate paths • Subsystem and/or component failure • Protocol and/or transport failure
Systems Management	• Asset bootstrapping (testing using reference agent) • Gateway and remote agent operation and ISMP message delivery • Remote configuration and upgrades of ITCR and ITCM assets via various transports • File Transfer/Distributions to one/many remotes • SW and configuration kit management such as file loads/activations and command policies • With locomotives at speed and stationary • Out of band access to assets

Test Scenario	Description/Conditions
Stability	• Soak testing and confidence testing • Functionality: SW robustness, effects of long-term message load, effects on system or system elements such as radio network operation • Performance: effects on message success rate and KPIs such as latency and throughput
KPIs	• Message latency • Data throughput and message rate • Above functionality with/without additional PTC traffic load • High asset density

ILT covers all levels of the test scenarios noted in Table 8. In terms of pure test case quantity and the direct validation of PTC communication system functionality, CTT is largely a subset of ILT. However, priority at CTT is put on test scenarios that are unique to a realistically fielded environment. As such, CTT generally expands on that applicable testing conducted in the lab. CTT covers many of the scenarios in Table 8 with specific interest in scenarios where OTA testing adds value, including the following:

- Realistic configurations and environment:
  - OTA RF links: antennae, propagation, and susceptibility to external noise
  - Tests with real locomotives at speed (up to 107 MPH)
- Mobility testing:
  - Locomotive base-selection
  - Base hand-off/over of locomotives
  - Messaging without base coverage and with or without alternative IP paths
  - Messaging at speed including direct peer-to-peer (PTC functionality)
- ETE tests in a field environment where many elements of the PTC communication system need to work correctly:
  - As such, some elements are directly validated while many other elements are indirectly validated

- A specific example is remotely upgrading various radios' SW over the 220 MHz RF link while normal messaging is also occurring over the 220 MHz RF link in the background throughout the system

3. System integration test methodology

3.1 Test planning

At the beginning of each product release cycle, MCC's IWS develops a high-level test plan, which is used to drive the following:

- Test case development and prioritization. Test cases themselves are used to drive test procedures as well as test tool development and optimization, all requiring preparation time. This "prep" time is fundamental in order to refine testing of new system functionality and increase test efficiency going into the next testing phase.
- A detailed tactical plan and schedule that identifies specific resource utilization (for example, test beds and personnel). Planning meetings are held on an ongoing basis throughout the testing of the product release cycle to further refine the schedule and resources in order to execute the full IWS test cycle.

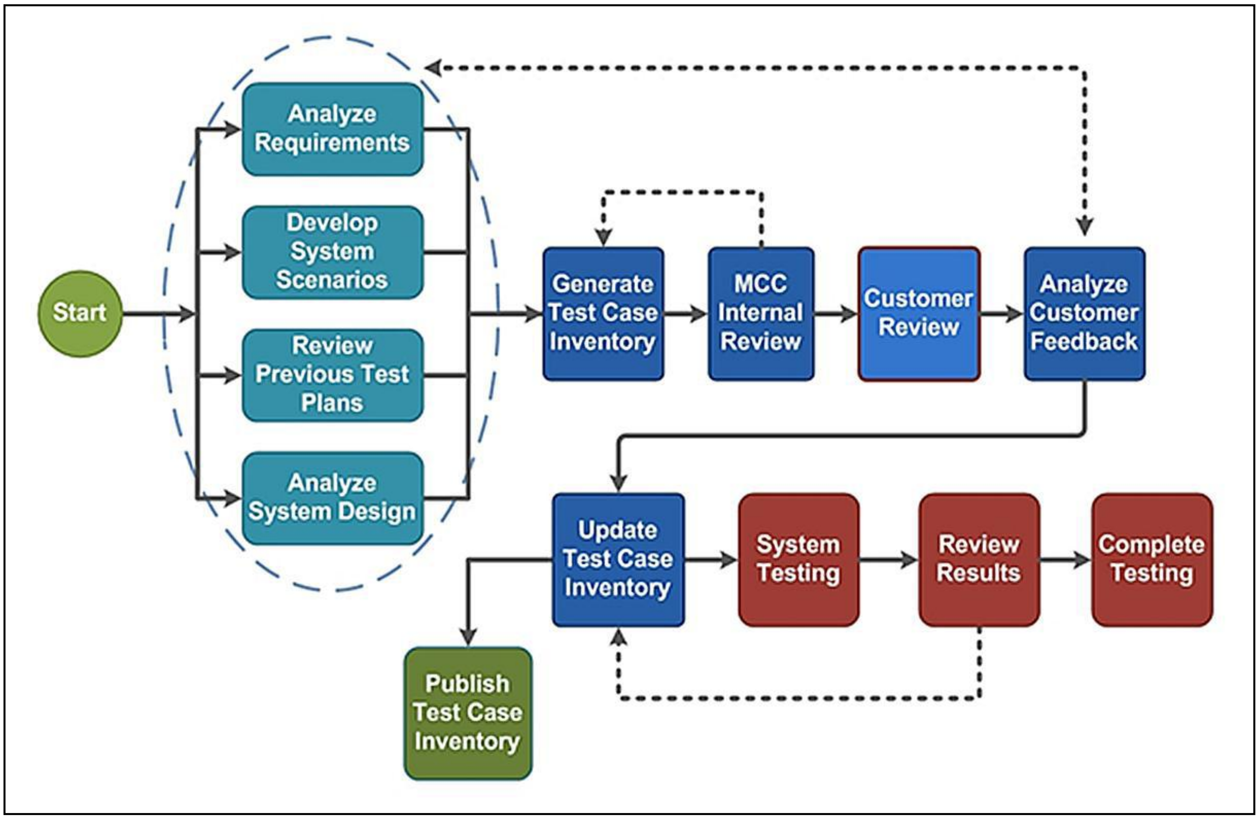
3.2 Test case development

Test cases are specific descriptions of tests to be performed. Typically, test cases are derived from test scenarios, which themselves are derived from use cases. This process is in the context of system requirements, which generally are also derived from use cases. That said, IWS test cases are developed using a variety of methodologies and sources, which include the following:

- Requirements analysis
- Use cases and scenarios
- Architecture and subsystem design documents and technical specifications
- Customer input and review
- Previous radio system test plans
- Prior experience with communication system development and testing
- Deep understanding of the communication system architecture and design

The list of test cases for each stage of integration undergoes various levels of internal review as it is developed. Once completed, the test cases and associated test configurations have undergone a more formal review with audiences that may be both internal and external to MCC. The IWS test case development process is outlined in Figure 8. (Note that not all of the methodologies listed above are included in Figure 8.)

Figure 8: Test case development and review



Following is a summary of the steps in Figure 8:

1. Using a number of methodologies, an initial test case inventory is created by MCC's IWS
2. The initial test case inventory goes through MCC internal reviews and is updated and refined as needed
3. After the internal MCC review process is completed, the test case inventory is sent out for customer review
4. Customers provide feedback to MCC in a timely manner generally in the form of written responses, conference calls, and face-to-face discussions
5. MCC reviews the customer feedback
6. Based on the review of customer feedback, the test case inventory is updated as needed and then published as the baseline set of test cases

7. As testing proceeds and results are analyzed, existing test cases may be refined and new test cases may be added

3.2.1 Requirements mapping

Each test case in the inventory is mapped to one or more PTC communication system requirements, derived requirements, or identified as good test practice (for example, basic system stability). Depending on the nature of the test cases and associated requirements, some test cases directly validate requirements while other requirements are indirectly validated. In some instances, multiple test cases may be needed to validate a single requirement. Conversely, a single test case is sometimes sufficient to validate multiple requirements. This is a many-to-many mapping. As detailed in subsequent subsections, each test case contains a reference to the designated requirements to which the test case applies.

Creating test cases that map to PTC system requirements requires an associated analysis. This analysis focuses on identifying which requirements are appropriate for testing at the system level. The following considerations are used for guidance as part of this analysis:

- It is not possible to validate the requirement at the subsystem test level
- The requirement can be evaluated using a system-level test
- The requirement covers the interaction of two or more subsystems

After the requirements driven test cases have been developed an ad hoc process is used to expand the test case inventory. This may include specific scenarios of interest to customers, corner cases, and tests that target derived requirements or specific design elements.

3.2.2 Requirements validation

For successful completion of ETE system tests, many system elements and subsystem components need to function properly. For example, within IWS testing, to send a message from a BO application to a simulated WIU endpoint over the 220 MHz radio network, multiple messaging SW components as well as the radio air interface need to function properly. Though such a test case may be specifically designed to directly validate message delivery from the BO application to the simulated WIU, in the process of running the test, other communication system elements are indirectly validated.

Direct validation of an ITC requirement is a necessary step towards complete validation of the PTC communication system requirements of a particular release. Integration testing as a whole will directly validate a limited percentage of ITC requirements where ILT will directly validate some and CTT validate some more. However, when integration testing is combined with the testing performed at the subsystem level, complete validation of the ITC communication system requirements is achieved.

3.2.3 Test case priority

Though an extensive inventory of integration test cases can be generated, in practice, the depth of integration testing must be balanced against an assessment of program constraints and risk. Additionally, in reference to assessing product performance, some test cases provide higher value than others.

Classifying the priority of specific test cases requires analysis and judgment. The considerations in determining test case priority include an evaluation of the following:

- Functionality deemed most mission-critical
- High-risk system functions, features, and subsystems generally prioritizing the testing of system elements that:
  - Are least tested
  - Are least understood
  - Are most complex
  - Have the most dependencies
- Functionality and features that are product defining (but may not be mission critical)
- System functionality and features that undergo the highest levels of usage
- How much value a test case provides

Some test cases can be classified as "high value", which in turn increases the priority of such test cases. A high value test case is one that yields a relatively large amount of system performance or functionality information and/or validates multiple requirements. High value does not necessarily equate to a P1 test case, as there may be considerations as noted above.

As a guide during test execution, ILT and CTT test cases are prioritized to the three levels indicated in Table 9.

Table 9: Test case priority levels

Priority Level	What it does and what prior coverage does it have	Effect of Failure
P1	• Tests an essential product feature • Tests a basic element of functionality • Tests functionality or performance that cannot or has not been tested in prior test phases	• Indicates a critical fault • Makes a large portion of the system unusable or an essential feature inoperable • These tests must be passed to exit the IWS test phase • These tests should be run early in the test cycle
P2	• Supports testing of important usage scenarios • Tests features or functionality not well tested elsewhere	• Indicates a loss of important functionality • Does not significantly interfere with further testing and/or there may likely be workarounds that can address the loss of functionality • A majority of these tests must be passed to exit the IWS test phase (as defined by specific exit criteria)
P3	• Does not directly test core functionality • Tests features that have been more extensively tested in prior test phases	• Indicates a slight loss of functionality • Has only minimal impact on further testing • These tests are not part of the exit criteria for the IWS test phase

3.2.4 Test case naming

Each test case has a unique name. The test case name contains the requirement ID and a sequential test case number for that requirement. In addition, the test case name contains additional information that may note what test platform or what type of test among other things. Some specific identifiers within test case names are as follows:

- Typically contains either an "ILT" or "CTT" prefix to indicate whether it is a test case for ILT or CTT
- Contains the integration test case ID, which is unique and identifies the focus of the test case
- Typically contains either the direct or derived requirement number validated by the test case
- May contain other identifiers such as "BC" for backwards compatibility testing

Typical test case names include the following:

- ILT_CMX-15_RadioAsset_Neg_TC04
- ILT_CR-4_MsgAsset_TC02
- CTT_ConfFR_SR-014_TC001_Cond0

3.3 Defect management

Through the integration testing process, which includes ILT and CTT, defects and performance deficiencies in the communication system are uncovered, which drive design changes and software updates. It is important that any such defects be correctly identified, recorded, and analyzed.

3.3.1 MCC defect tracking and correction (internal process)

MCC's IWS manages communication system defects using a defect management tool that allows defects to be entered into a database, assigned an owner, and tracked for progress towards resolution. The process of entering a defect into the database and tracking it to closure is referred to as the defect lifecycle. Defects could be either HW or SW and the same defect lifecycle process is used for ILT and CTT. The MCC defect lifecycle process is outlined as follows:

1. A potential defect is identified through testing or its subsequent analysis.
2. The defect is entered into the database with information detailing the conditions under which the defect is manifested, a proposed priority and severity, and a proposed defect owner.
3. The defect is triaged by a cross-functional team. This is a process by which defects are reviewed and assessed for priority, severity, and frequency of occurrence. The higher the priority, the more attention towards resolution a defect will obtain.

4. After triage, defects are investigated to ensure they are not duplicates of previously entered defects and to establish that they are valid.
5. Once the scope of a defect and expected resolution effort is understood, a plan for corrective action is developed and resources are allocated as appropriate.
6. Once a defect has been identified as fixed, the resolution is validated, typically by the test team.
7. After the defect resolution is validated, the defect is closed.

3.3.2 Defect reporting and resolution information

The defect lifecycle process is most efficient and effective when all parties involved are diligent about providing good detail with timely communication. Following is a list of key information that should be included when entering defects into the database.

The following highlights information to be entered into the defect tool by the **originator**:

- SW version(s): include any special files, component loads (for example, FPGA, DSP, etc.), or scripts that were introduced after the SW was built.
- HW version(s).
- Specific test case(s) under which the defect was identified.
- Note the conditions. For example:
  - Is the issue seen with all radios?
  - Which test configuration?
  - What situation or actions led to the issue?
- Record the frequency of occurrence of the issue:
  - Always?
  - Periodically?
  - Occasionally?
- Is the issue repeatable? How easily?
- Attach files, documents, and so forth as needed.

The following highlights information entered into the defect tool by the **resolution provider**:

- How was the issue resolved?

  - SW or HW change?
  - Documentation update?
  - Requirements clarification or change request?

- Is the fix a solid fix or a temporary workaround?
- What SW modules, sections, etc. are affected by the change?
- What HW platforms does the change impact?
- What unit tests or analysis was applied for initial verification of the fix?

3.3.3 Defect severity

The severity of a defect measures the effect a defect has on the ability of a product or system to meet its defined requirements. For integration testing, MCC uses four levels to identify defect severity. This rating scheme is summarized in Table 10. Along with the defect's frequency of occurrence, severity is used to assess the priority level that the defect will receive.

Table 10: Defect severity definitions

Severity Level	Value	Description
Critical	1	• Causes complete or widespread loss of functionality within the system • Significant performance issues that are unacceptable for live operation • No acceptable workaround exists
Major	2	• Causes widespread loss of functionality or non-use of key product features • A work around may exist but its use is unsatisfactory
Minor	3	• Failure of non-critical aspects of the system, or functional problem with little impact on overall system operation • There is a reasonably satisfactory workaround
Low	4	• Defect of minor significance • Minimally impacts system functionality • A work around exists, or if not, the impairment is slight

3.3.4 Defect fix validation

During the defect resolution process there are multiple validation stages. These are summarized in Table 11.

Table 11: Validation test stages for defects identified during ILT and CTT

Validation Stage	Description	Tester
Unit Test	Testing performed on the design element to which the defect fix was applied	Element designer or team within the applicable subsystem workstream (MWS, RWS, or SMWS)
Module Test	Testing performed on a sub-element of one of the PTC communication subsystems (ITCR, ITCM, or ITCSM)	Module design or test team within the applicable subsystem workstream (MWS, RWS, or SMWS)
Subsystem Test	Testing that is performed on a PTC communication subsystem (ITCR, ITCM, or ITCSM) as separate entities	Applicable subsystem workstream's test team (MWS, RWS, or SMWS)
System Test - Lab	Testing of the integrated PTC communication system using the lab integration test bed	Integration Test (IWS in ILT)
System Test - Field	Testing of the integrated PTC communication system using the closed track test bed	Integration Test (IWS at CTT)

Depending on the nature of the defect, one or more of the validation stages prior to system test may not be warranted. For system test defect validation the following process is applied:

1. The SW or HW, to which a defect fix has been applied, is installed in the appropriate IWS test bed
2. The conditions under which the defect was discovered are replicated and test cases are executed to validate the defect has been resolved
3. After the defect fix has been validated, regression testing is performed to evaluate the impact the resolution might have had on other associated system functionality

3.4 Test case and defect tracking

To track and manage its test process, MCC's IWS uses a test management tool called Hewlett Packard Quality Center (HPQC) that provides the following features:

- Tracking of and traceability between requirements and test cases
- Tracking of test progress
- Database for test results storage and report generation
- Central management of manual and automated testing assets
- Defect management including reporting and tracking of resolution

3.4.1 Test case information

There is a variety of information associated with each test case that HPQC stores, tracks, and manages, including:

- Test Case ID: A unique abbreviated ID, which help identifies the focus of the test case.
- Requirement number and linked requirements: A number is usually included in the test case name that identifies the requirement(s) that is (are) tested by the test case. Also, all applicable requirements are linked.
- Priority: Indicates the importance of a test case. Test cases can be given a priority of P1, P2, or P3 with P1 being the highest priority.
- Description: A description of the actual test case including what functionality it is testing.
- Procedure: Outlines the steps involved in executing the test case.
- Expected result: Description of the expected test outcome for a passing result.
- Test Bed: Identifies the test bed and/or test configuration used to implement the test case.
- Test Setup and Configuration Instructions: Notes any special equipment or configurations that are required.
- Additional information providing traceability of the test case to a specific specification, derived requirement, or other document.
- Status: The result of the test case over time, which can be NO RUN, INCOMPLETE, PASSED, FAILED, or BLOCKED.
- Linked Defect(s): Any defects that contribute to a non-passing result of the test case are linked.

3.4.2 Progress tracking and results reporting

As testing proceeds and test cases are performed, HPQC stores the test results and notes which test cases have been completed. This allows test progress to be tracked against PTC communication system requirements on a daily basis. Metrics such as test completion rate can be defined and used to indicate trends in test progress. At any point in the process progress reports can be produced.

3.4.2.1 Test case status

Test case status starts out as NO RUN. A test case is evaluated for execution and if the test case cannot be run due to a known defect or lack of functionality, then the status is changed to BLOCKED. After execution of the test case, the status is changed to either PASSED or FAILED, depending on the outcome. A status of INCOMPLETE denotes a test case that has started but is still in progress or its results are being analyzed.

Test cases are conducted multiple times as the subsystem SW moves through the development cycle. As new code is released for the various system components, ILT is moved through different test sprints and as a result, the pool of test cases are carried out again on the updated PTC communication system. As testing progresses, it may be determined that some test cases may no longer be needed and are therefore retired while new test cases are added as required to cover new functionality.

HPQC keeps the history of the status of all test cases. For reporting purposes, the status of the last run is typically reported. For example, if the test case had passed 10 times and failed the last run due to a new defect, the status is changed from PASSED to FAILED and the new defect number is referenced.

3.4.2.2 Defect status

As noted above, the associated defect number(s) is (are) referenced for any FAILED or BLOCKED test case. Though defects can be entered into HPQC by both the test group (for example, IWS) and the development teams (for example, the subsystem workstreams), defects can only be closed by the test team who discovered the defect once they have validated the fix.

HPQC allows the tracking of much of the defect information detailed in Section 3.3. It should be noted that the status of a defect can be any of the following:

- NEW: Defect has been found in IWS but not triaged, which is typical of “just-found” defects prior to the first triage meeting following their discovery. Defect is assigned to triage.
- OPEN: Defect has been triaged and assigned to a team or individual to fix.
- FIXED: Defect has been corrected but has not been validated by IWS. Defect is assigned back to team or individual who initially reported it.
- CLOSED: Defect’s fix has been validated by IWS and therefore closed.

3.4.2.3 Test reporting

IWS produces two types of reports during the testing of a particular product release cycle. There are weekly status reports as well as a final end-of-release R1.x report.

Weekly reports provide status on the following:

- Aggregate test case status, which yields the number of test cases that are PASSED, FAILED, BLOCKED, NO RUN, etc.
- Issues that could pose risk to project schedule
- Planned versus actual test case burn-through
- Aggregate defect status, which yields the number of defects (by priority) that are NEW, OPEN, FIXED, and CLOSED
- Defect trends
- Complete defect list for integration-found issues

The final R1.x report provides the following information:

- Test execution summary for all test cases
- New featured test results
- A complete defect list for the entire product release cycle
- Testing that was not performed
- Known open issue list

Appendix 0 shows examples of plots that are included in the IWS reports.

3.5 Test categories and types

MCC's integration activities include many tests. These tests are sorted into test categories, which are often used to organize test suites and tactical (daily or weekly) planning for test implementation. These test categories may be based off of the type of testing to be done. For the purposes of IWS testing, the subsequent subsections detail the salient types of testing and Table 12 defines the test categories that IWS has used to organize test implementation and planning activities. The test types also build off of the test approach, that which is specifically detailed in Section 2.4.

Table 12: Test category definitions

Term	Definition
Sanity Tests	An initial set of tests designed to perform basic validation of the test environment and test candidate. These tests tend to be cursory in nature and focus on proper test bed setup and basic system stability.
Basic Functionality Tests	Tests to determine whether or not a specific system element, which is essential to fundamental system operation, is working. Functional tests are typically pass/fail.
Feature Tests	Tests of system features or multi-faceted functionality under nominal conditions. Feature test cases tend to be more complex than those for testing basic functionality and often include multiple parametric elements. Examples of such tests include: • Message rerouting over alternate transports due to poor availability of primary paths. • Virtual wayside messaging

Term	Definition
Failure or “Negative” Tests	Tests of how the system reacts when the user specifies system functionality that is incorrect or not supported. Examples of such tests include: • Assuring no message delivery if the message QoS specifies a class of a transport that is unavailable • Assuring the user receives an error response (and that the system does not conduct the action) if the user specifies an invalid ISMP parameter in a file distribution request.
Benchmark	An established level of overall system functionality and/or performance for a specific product release and conditions.
Baseline	An established level of system functionality or performance under default conditions. Some examples include: • Message latency between a wayside and locomotive over a wired Ethernet connection. • Pass/fail rate of basic functionality test cases in a small system with radio.
Performance Tests	Tests to determine how well some aspect of the system behaves under a particular workload as compared to a defined set of metrics. Performance metrics are quantitative in nature. Some examples are message latency and data throughput.
Stress and Boundary Tests	Tests of the system under multiple simultaneous conditions and/or at the edges of specified system operational performance. For example, remote to BO message latency during high message traffic volumes and while one or more transport paths are intermittent.

3.5.1 End-to-End (ETE) testing

ETE test cases look at system performance and functionality with message traffic between two or more end points. An example is testing message latency between a remote and the BO.

The use of ETE testing is widespread throughout integration test. This is due to the fact that integration testing focuses on system-level behaviors that encompass multiple subsystems and components of the PTC communication system. In order to achieve a positive outcome for a given ETE test scenario, many of these elements within the communication system need to work correctly. Therefore, in the process of executing ETE testing to directly validate system-level functionality, the proper functionality of many elements within the system is also indirectly validated.

3.5.2 Regression testing

Regression testing involves rerunning tests that have previously been performed either in a prior stage of the program or within the current stage. Regression testing comes into play whenever there are product design changes or software updates for fixes or to add new functionality. As such, some level of regression testing is expected at each code drop or subsystem release into integration. Also, through the integration testing process, defects and performance deficiencies in the communication system are uncovered, which drive design changes and software updates. Regression testing is needed when these changes are introduced to validate fixes and to assure product operation that existed prior to an update or change still exists afterwards.

Regression testing starts with validating the effect of the specific changes or updates and then expands to include other system or design elements that may have indirectly been impacted by the changes. For any particular design change or software update, the depth of regression testing to be performed is determined by an analysis of the changes. Both the test group and the development team participate. Based on their analysis, previous test results, and an in depth understanding of the design, a judgment is made as to the extent of regression testing required.

3.5.3 Backward compatibility testing

Backward Compatibility Testing (BCT) validates proper PTC system operation while the SW versions in the various assets and components are systematically changed in various combinations. For example, initial testing may occur on the system when the BO and most remotes might be loaded with the latest SW while a few other remotes are loaded with a previous version of ITCM SW. The same testing is then done on the system where the BO is loaded with a previous version of ITCM SW but the remotes are all loaded with the latest SW. The process repeats where each time a new permutation of SW versions on various assets are loaded.

Backwards compatibility first became a requirement of the PTC communication system starting with R1.1. During R1.2 testing, “N-1” testing was undertaken to assure compatibility between R1.2 and R1.1. However, the ultimate requirement of backwards compatibility is “N-2”, which means the most recent three major releases must work with each other. Testing for full N-2 will start with R1.3 where its functionality must be compatible with R1.2 and both with R1.1. When R1.4 testing begins, compatibility must exist between it, R1.3, and R1.2. Under the N-2 requirement, R1.4 compatibility with R1.1 will no longer be necessary.

Implicit in the aforementioned, BCT refers to the compatibility between major product releases of the PTC communication system and assumes the latest applicable minor releases for each subsystem are in play. For example, when referring to ITCR R1.1 SW, it is assumed that the latest applicable minor release of ITCR R1.1 (which was v1.1.15.6) is being referenced.

As of this document’s inception, R1.2 has just been released. As testing progresses into R1.3, BCT will eventually focus on all of the following areas.

- SW upgrade process
- Peer to peer communications
- Inbound/outbound connectivity and mobility
- HRX failover
- SMS including RAD-SMS and MSG-SMS functionality
- Remote to office IP connectivity
- Inter BO scenarios

3.5.4 Stability and soak testing

Soak testing is a long duration test designed to evaluate the PTC communication system’s operational stability. Initially, soak tests are conducted under nominal traffic load conditions and without any intentional impairments such as component failures. Downstream, as time and resourcing permits, soak testing will include some level of impairments such as radio failures, message flooding, and large file transfers. Periods of system idle time will also be included.

A primary preference for soak testing is covering the most system-level functionality in a given set amount of time, which implies the following.

- The utilization of ETE message tests of varying sizes, rates, QoS parameters (such as priority, network preference, etc.), and types (for

example, normal PTC traffic as well as ISMP messages) to exercise the system
- The use of the most assets as possible which generally means use of the largest IWS test beds

Ideally, a soak test cycle will run for several weeks without interruption. However, this ties up a test bed for extended periods of time. As such, the duration of each soak testing cycle needs to be balanced against other test priorities. In a given product release cycle, soak testing typically starts out at approximately 2-4 days in duration and expands upwards of a few weeks as the product release's ready date approaches.

3.5.5 Key performance indicator testing

Key Performance Indicators (KPIs) are quantifiable metrics that reflect certain performance levels of the PTC communication system. Measurements of KPIs are used to validate some requirements. Additionally, KPIs provide a metric to compare various system configurations, software releases, and hardware versions. A list of PTC communication system KPIs are given in Table 13. All of these KPIs are expected to be tested at either the subsystem or system level.

[Portions of the following table have been deleted due to proprietary and confidential protections afforded under Exemption 4 of The Freedom of Information Act. The omitted information is a protected trade secret related to test results obtained from integration testing of proprietary software and hardware integration tests performed during development testing of pre-production radios related to the PTC 220 MHz radio communication system. Accordingly, this information is exempt from FOIA search and disclosure.]

Table 13: PTC communication system KPIs

Category	Description	Requirement	Notes
Message Rate	Back office message rate		• Measurement time is ≥ 8hrs
Message Rate	Mobile[i] message rate		• Measurement time is ≥ 8hrs
Message Rate	Wayside message rate		• Measurement time is ≥ 8hrs

Message Routing	Message routing updates		
Message Routing	Path updates		
Message Routing	Message path updates between offices		
Message Latency	Locomotive and Wayside segments		
Radio Network Latency	Time to shed lower priority traffic for higher priority traffic		• Based on QoS
Radio Network Latency	Latency for high priority messages (radio-to-radio over the air)		• Over the air = wired[ii] side of one radio to the wired[ii] side of another radio • Applies to message sizes ≤ 256B • Applies during Loco handoff between Bases
Radio Network Latency	Latency for high priority messages (application-to-application)		• Applies to inbound and outbound messages • Applies during Loco handoff between Bases • Does not include Wayside status related messages
Radio Network Traffic Load	System must support a minimum number of wayside beacon/status messages.		• Under the coverage of a single base • Loading model = PTC_Demand_Study_Version_03.xls

Category	Description	Requirement	Notes
Radio Network Traffic Load	Density of Wayside beacon/status messages.		• Under the coverage of a single base • Loading model = PTC_Demand_Study_Version_03.xls
Radio Network Traffic Load	Message traffic bit rate between Base and Locomotive.		• Either inbound or outbound message direction • Not including link level overhead
Radio Network Traffic Load	Operational PTC trains supported		• Under a single Base Station

i. A "mobile" is synonymous with a Locomotive in the PTC communication system
ii. "Wired" side of a radio means the back-end of the radio [that is, its Ethernet port(s)]

3.5.6 Manual and automated testing

IWS test activities include both manual and automated testing. Manual testing is necessary in the initial development and operation of most test cases, especially in the early product release cycles when both system functionality and test-control tools were less mature. This manual approach offers a deeper dive into new functionality and any potential issues, which enables a more in-depth understanding of both the system's operation as well as the capability and needed updates to test-control tools. In other words, it is an essential step in initiating test automation.

Over time, in terms of the maturity of the test case and test-control tools as well as the PTC communication system's functionality, many of the manual test processes and specific test cases are automated. Test automation addresses the goal of making overall testing more efficient, which is vital in allowing test coverage of the continually growing number of system functionality and requirements. The process of automation is facilitated by test-tool development as well as the scripting of both test cases and asset configuration. Section 5 discusses these concepts and the IWS test tools in more detail.

4. System integration test environments

As previously mentioned in the document, MCC's IWS test activities are conducted in two basic environments: one in a lab, the other in the field. The discussion of these platforms (ILT and CTT) in Section 2 and especially in Section 2.4 details their approach, test scenario scope, and how they fit into the overall PTC communication system test process. This section describes the IWS test platform setups and implementation including test bed configuration and infrastructure.

4.1 Test bed elements

The IWS test beds encompass elements that replicate or simulate as much as possible, the connectivity and data-traffic environment that the PTC communication system will encounter during real-world use. Although the ILT and CTT platforms are notably different (since the former is in a lab and the latter is in the field), they also share many similar elements. Following is a list of the salient test elements in the IWS test beds where those unique to either ILT or CTT are noted:

- PTC communication subsystems:
  - ITCR (Base, Locomotive, and Wayside radios)
  - ITCM SW
  - ITCSM SW
- Reference HW computers to host ITCM and ITCSM SW:
  - Single-Board Computers (SBCs) for remote sites
  - Servers for BO sites whether home/foreign, primary/backup, as well as either in-house at MCC or on-site at the CTT facility
- Alternate IP paths as transports including Cell, Wi-Fi, and for ILT only, Wired Ethernet
- EPSs for TMC, WIU, and BO applications:
  - In ILT, each remote EPS is located on a server located in the BO that directly connects to its remote communication site
  - At CTT, each remote EPS is collocated with its remote communication site on a unique SBC
- Test control including a script automation framework and results database connection
- Other Infrastructure:

- RF Network for 220, Cell, Wi-Fi, and GPS:
  - Cabled for ILT where some TBs include electronically controlled attenuators
  - OTA for CTT
- Ethernet backbone via various IP paths such as Cell, Wi-Fi, or for ILT only, Wired Ethernet:
  - Typically cabled RF or hardwired for ILT where test-control routes are physically isolated from system transports
  - Typically OTA for CTT either via Wi-Fi or Cell although base sites have hardwired backhaul
- Real railroad infrastructure at CTT only:
  - Locomotives at speed
  - Closed track loops
  - Fixed-site equipment housings and antenna towers

4.2 System integration test beds (SITB)

Multiple test beds have been built to support both ILT and CTT. Within either platform, the test beds have different configurations and numbers of assets, which provide the flexibility to manage multiple code drops concurrently and run a variety of test activities in parallel. The IWS test beds are summarized in Table 14.

Table 14: Summary of IWS test beds

Test Platform and Location	Test Bed ID	PTC Radios			Alternate Path Modems		Remote ITCM Servers	Back Offices	Reference Assets	Product Release Cycle[i]
		B	L	W	Wi-Fi	Cell				
ILT (MCC)	SITB1	2	6	15	21	21	21	1	1	< R1.0
ILT (MCC)	SITB2	1	2	3	5	5	5	1	1	< R1.0
ILT (MCC)	SITB3	2	2	4	6	6	6	1	1	< R1.0
ILT (MCC)	SITB4	2	2	4	6	6	6	1	1	< R1.0
ILT (MCC)	TTTB	3	3	3	6	6	6	2	1	R1.1
ILT (MCC)	SSTB	2	2	2	4	4	4	1	1	R1.2
ILT (MCC)	SITB7	1	1	1	2	2	2	1	0	R1.3
CTT (MCC)	CTTB	2	2	4	6	6	6	2	0	R1.0
CTT (TTCI)	TTCI[ii]	3	3	7	10	10	10	2	1	< R1.0

i. Notes the first product release cycle that the test bed was used for
ii. Commonly referred to as simply “CTT”

The differences between ILT and CTT test beds are further explained in the following sections.

4.3 SITB configuration in ILT

The ILT environment consists of test beds that combine the PTC communication subsystems (PTC 220 MHz radios, ITCM and ITCSM SW), reference HW platforms (remote and BO) into an accessible and testable lab-based integrated PTC communication system. Each SITB in the MCC integration test lab is in a rack environment as shown in Figure 9.

Figure 9: System integration test bed

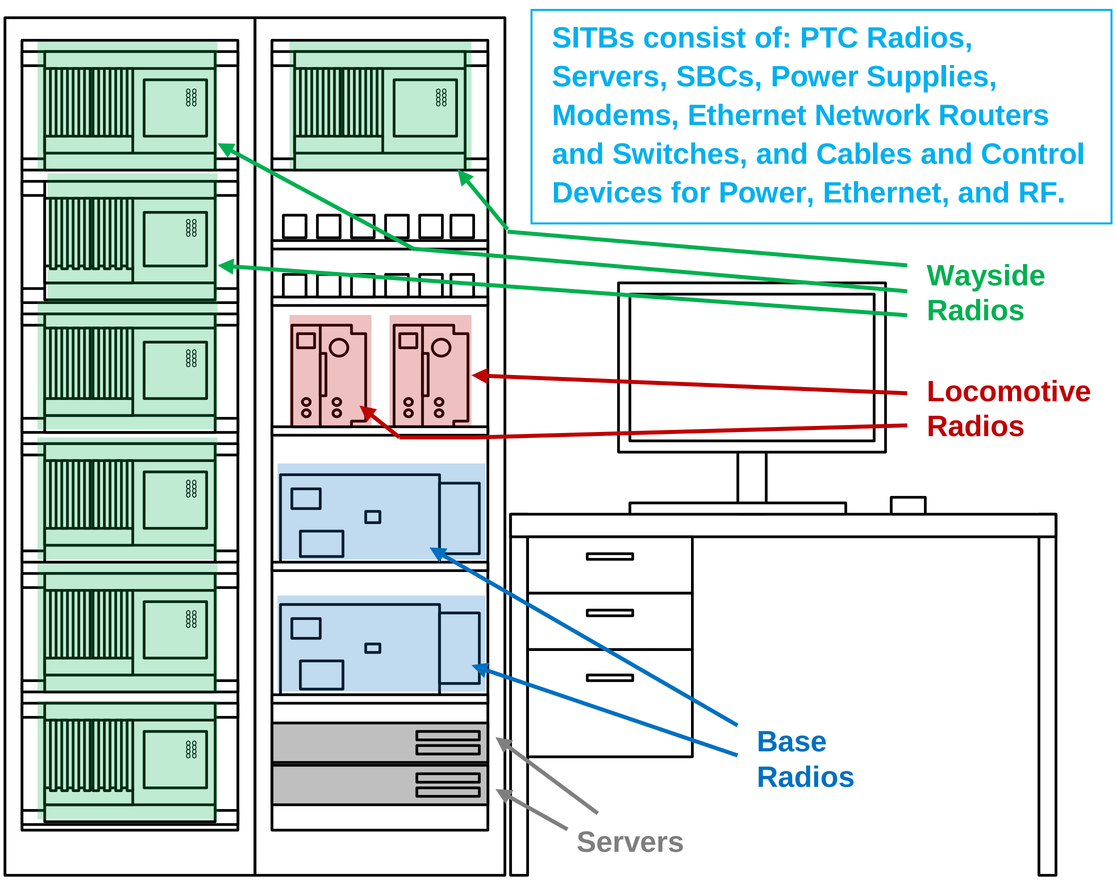
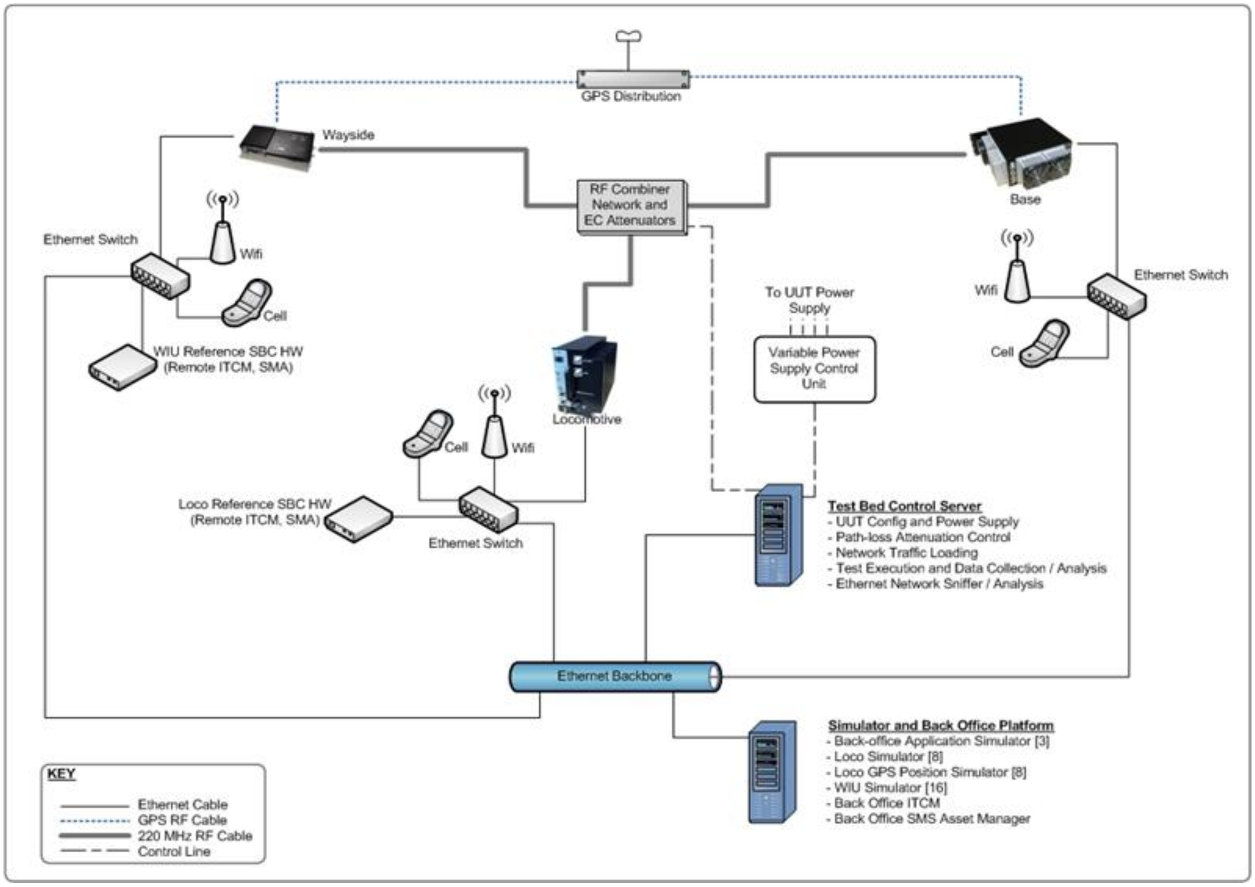


Figure 10 shows a basic block diagram of the SITB configuration where only one of each communication site is shown. More communication sites would simply be added by cabling into the RF combiner, Ethernet backbone, and GPS distribution networks. Each SITB consists of the following major elements, which are described in the following subsections:

- General infrastructure
- PTC 220 MHz radio network
- Alternate IP paths
- ITCM and ITCSM SW and their reference architecture
- Test control

Figure 10: SITB configuration for ILT



4.3.1 SITB infrastructure

The SITB infrastructure includes the following:

- RF interconnects and combiner/distribution networks for 220 MHz, Cell, and Wi-Fi RF paths
- Electronically controlled attenuators for the 220 MHz mobile-radio-to-other-radio RF links
- Electronically controlled power supplies for all components including PTC radios, reference HW SBCs, Wi-Fi and Cell modems

- GPS signal distribution network and amplifier for the fixed-site radios
- Ethernet backbone network that interconnects all Ethernet-enabled components and also allows for electronic control of each transport's physical connection at each individual communication site

4.3.2 SITB PTC radio network

The PTC 220 MHz radio network used in each SITB is made up of the following where the quantity of radios in a specific test bed is identified in Table 14:

- Wayside Radio(s)
- Locomotive Radio(s)
- Base Station Radio(s)
- 220 MHz RF network including electronically controlled attenuators

The RF ports of the radios (Tx and Rx) are interconnected using the 220 MHz RF combiner network. This allows each radio to communicate with other radios in the SITB network directly over the PTC communication system's 220 (Radio) transport. There is a high-power fixed attenuator in line with each radio's RF port. This attenuator, plus the RF cabling and combiner network, presents a fixed amount of path-loss for each radio-to-radio path. In the SITB "baseline" setup, this fixed loss amounts to each radio easily hearing all other radios in the SITB network.

As noted in the previous subsection, some test beds in the integration lab also include electronically controlled attenuators in their 220 MHz RF network to specifically control each mobile-radio-to-fixed-site-radio link in a real-time manner. This dynamic varying of the 220 MHz RF link allows realistic simulation of locomotive movement in a lab environment.

4.3.3 SITB alternate IP paths

Along with the 220 transport, each SITB also has other transports in the form of alternate IP paths including Cell, Wi-Fi, and hardwired Ethernet. The primary purpose of the hardwired Ethernet path is to validate SITB operation and establish performance baselines. The Wi-Fi and Cell connections are used (along with the 220 transport) to validate various message routing and transport-failover scenarios.

4.3.4 SITB ITCM and ITCSM SW's reference architecture

Each SITB contains computers that host ITCM and ITCSM SW in the BO as well as each remote.

In the BO, each SITB contains multiple servers that are running the RHEL OS. One of these servers hosts the BO ITCM SW and as such, is typically known as “the BO” or the “BO server” (even though other servers are also located in the BO). Another RHEL server in the BO contains the ITCSM SW such as the Gateway, also known as the SMS AM.

Each remote communication site contains a reference HW SBC that hosts the remote ITCM server and its ITCSM Agent, also known as the SMA specific to ITCM. Some communication sites (typically one per SITB) also contain an SBC that hosts an ITCSM reference asset for SMS-specific testing.

Each remote radio connects to its own ITCM SBC through an Ethernet switch. Typically, all Base radios in an SITB connect back to the SITB's ITCM BO server. In cases of federation testing, however, a Base radio may be tied to a foreign BO's ITCM and EPS servers.

4.3.5 SITB control server and simulators

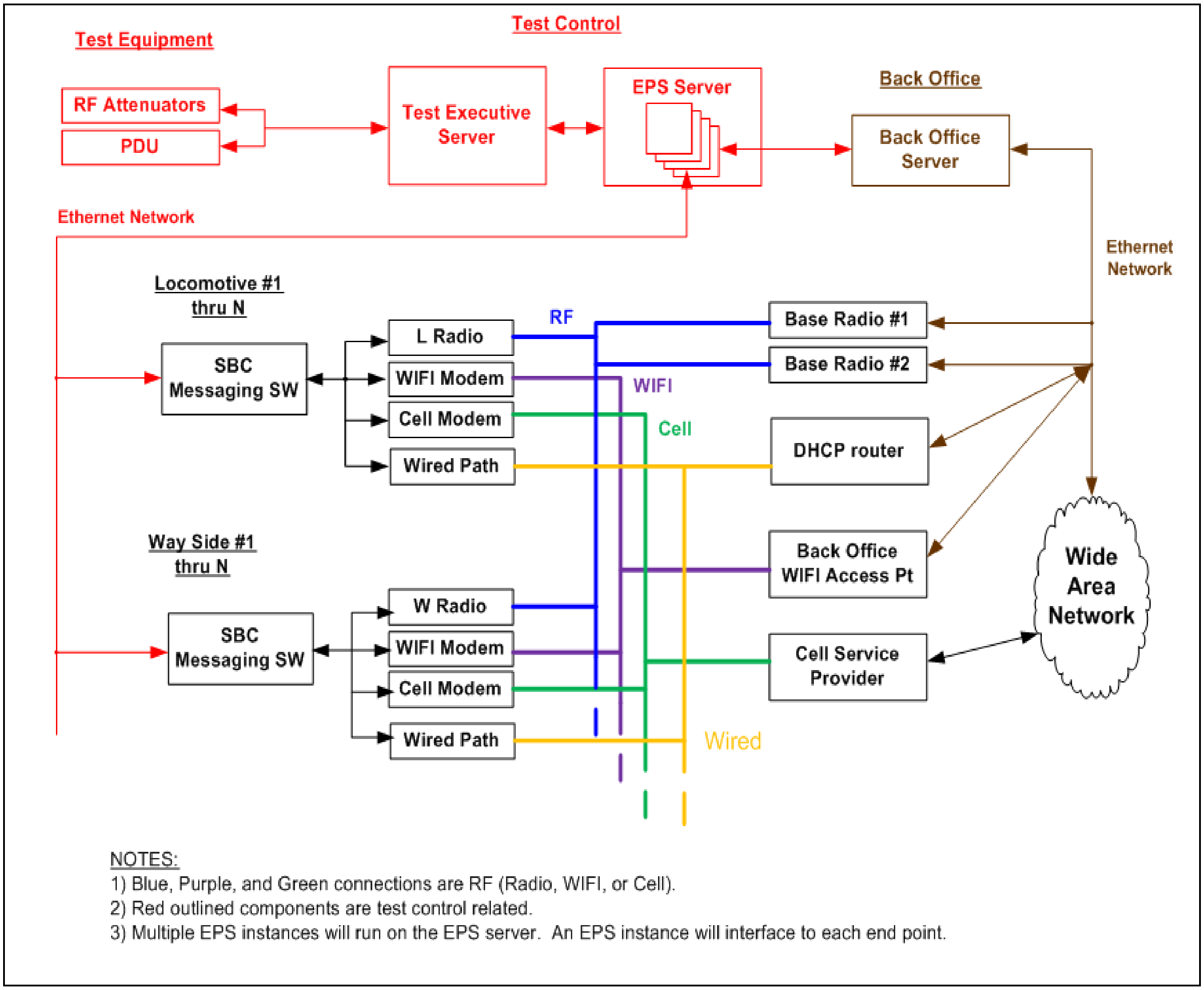
One or more computers are used to host a variety of test-control applications and simulators. Along with ITCSM SW, the second RHEL server in each BO (called the “EPS server”) contains all SITB assets' EPS instances and also serves as the primary ITCM deployment host for the ITCM assets in that particular SITB. The EPS instances simulate BO, WIU, or TMC applications as applicable.

Additionally, each SITB also contains at least one Windows-based server that provides test control and is generally referred to as the “Test Executive server” or “TE server”. The TE server hosts a variety of tools and applications that provide test bed control, data collection, and test automation for the entire SITB. This includes the following:

- Test execution and data collection/analysis
- Test suite automation and control SW
- Subsystem and asset configuration including power supply control
- RF path-loss attenuation and physical-layer transport control
- Network traffic loading
- Ethernet network sniffer/analysis

This overall IWS test-control implementation, which includes the various servers (TE and EPS) and test tools/applications, is collectively known as the test-control system and is further discussed in Section 5. Figure 11 shows the ILT SITB setup with the essential tools of the test-control system.

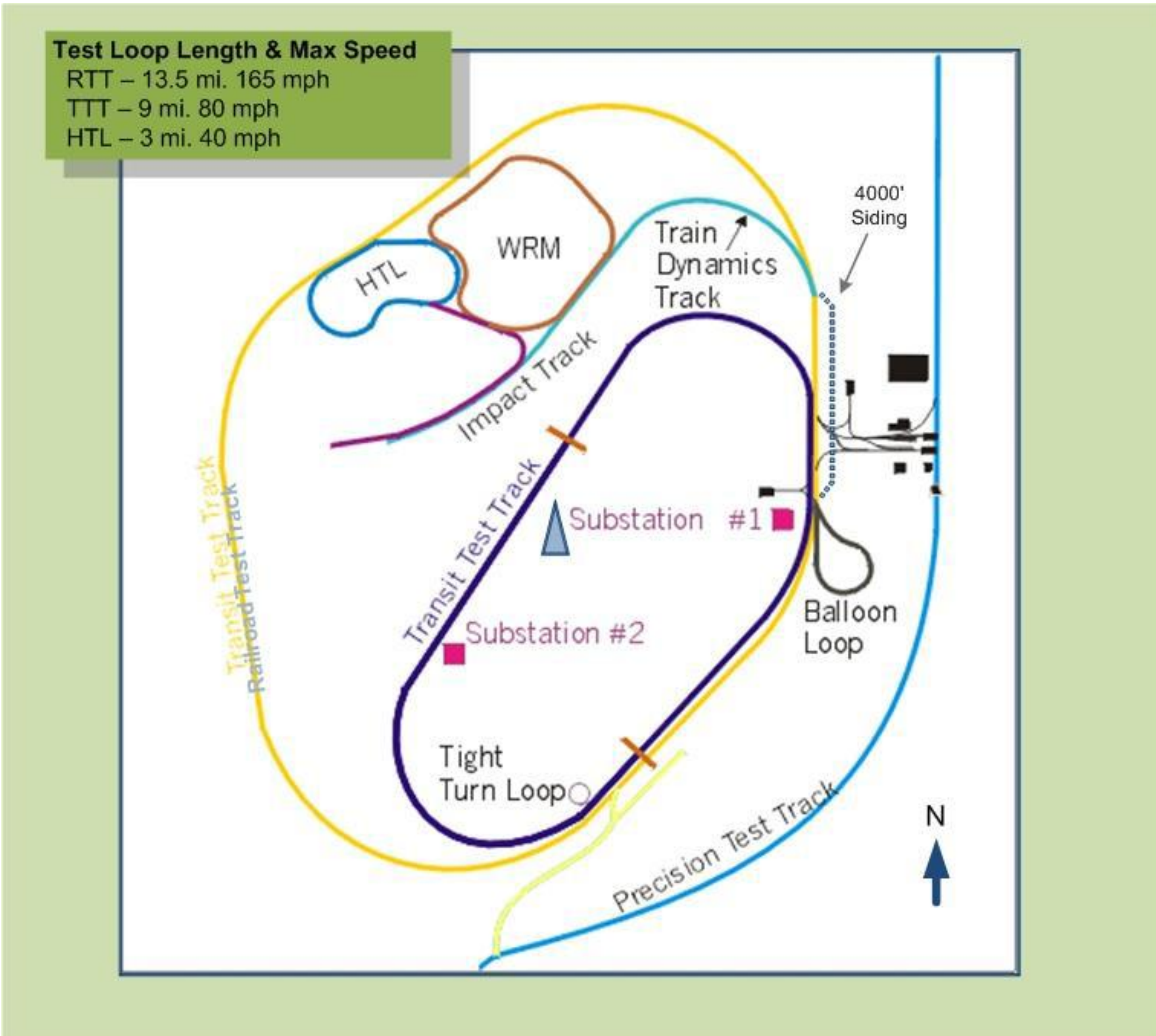
Figure 11: SITB showing test-control



4.4 CTT field environment

MCC's CTT setup entails the installation/maintenance, operation and testing of the integrated PTC communication system at the Transportation Technology Center (TTCI) located near Pueblo, Colorado. This facility has multiple test track loops "closed" to revenue railroad traffic, hence the name Closed Track Testing. An overview diagram of the CTT tracks is shown in Figure 12. The primary test track used for CTT is the Railroad Test Track (RTT), which is a 13.5 mile loop that supports locomotive speeds in excess of 100MPH.

Figure 12: TTCI test track diagram



The wayside communication sites are housed in some of the 12 weatherproof equipment enclosures spaced approximately every 6000 feet around the RTT. Each enclosure is wired with electrical power and is adjacent to a pole where the 220 MHz antenna is mounted.

Similarly, there are three towers where the Base antennae are mounted at the CTT facility. Neighboring housings complete with electrical power accommodate the Base radios and their associated setup, which includes backhaul access (via Wi-Fi or hardwired Ethernet) to the on-site BO location. Two of the three base housings are climate controlled. The third base housing, however, like the wayside bungalows, is not.

Additionally, the CTT facility has fairly robust cellular coverage to all of the remote communication sites as well as most of the RTT. The Wi-Fi coverage is reasonable to many of the communication sites.

The following subsections add more detail on the IWS CTT setup and environment. The reader is also encouraged to look at reference documents [3] and [7] for additional details as needed.

4.4.1 RF environment

The RF propagation environment at TTCI is relatively benign. The local topology is relatively flat with little elevation difference around the RTT. Additionally, there is little clutter and vegetation of major significance. Moreover, much of the RTT and other test tracks are on a raised bed, which further allows all communication sites to have clear line-of-sight to many of the other sites.

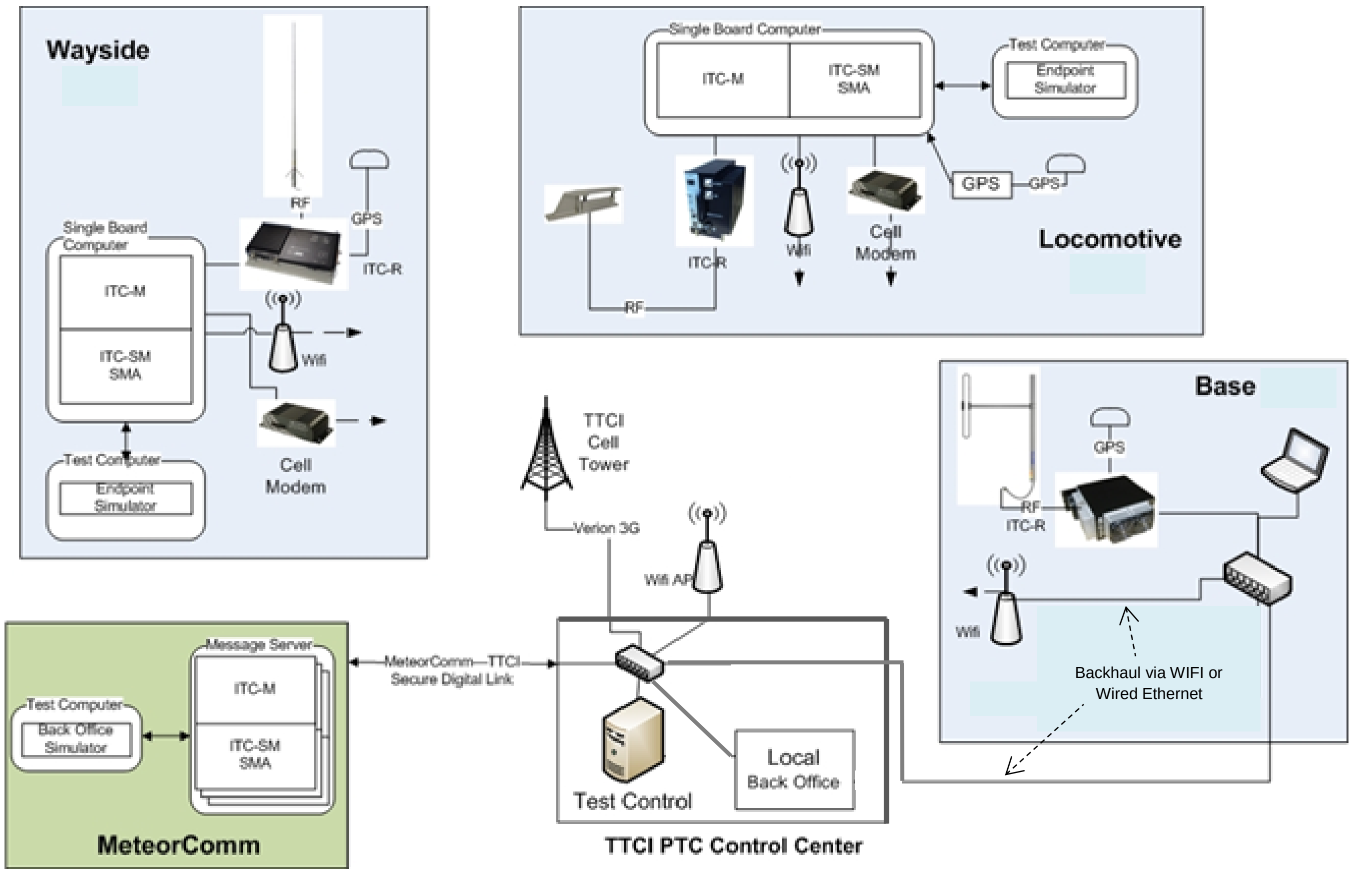
This CTT environment supports testing under controlled field conditions, which provides a near real-world platform for testing the integrated PTC communication system. Since the area is RF-friendly and relatively small as compared to a full PTC market deployment, it is a stepping stone for going from the ILT environment to various real-world “open” track environments, which collectively have many more data-communication obstacles.

Given the smaller area of the CTT environment, attenuation has been added to many of the PTC 220 MHz radios in-line between their respective antenna and RF port. Similar to the ILT SITB setup, this includes a high-power fixed attenuator. However, it may also include a variable attenuator that is used to “dial in” the desired coverage for any number of specific test scenarios (for example, to limit a Base radio’s coverage to only a portion of the RTT).

4.4.2 Test bed configuration at CTT

Figure 13 shows a basic block diagram of the CTT configuration displaying only one of each type of communication site. Conceptually, more communication sites could be added as needed to make up the actual totals deployed at the CTT facility.

Figure 13: CTT configuration



Note the fair amount of consistency with the ILT setup shown in Figure 10 where wayside and locomotive communication sites include their associated PTC radios, both Cell and Wi-Fi modems, and an SBC to host the remote ITCM and ITCSM SW. Conversely, there are some notable differences between the CTT and ILT test configurations (the first two bullets have been noted earlier in the document):

- CTT has real OTA RF links for the 220, Cell, and Wi-Fi paths
- CTT has real railroad and terrestrial communication infrastructure
- At CTT, each remote EPS is collocated with its remote communication site on a unique SBC
- The test-control paths and actual PTC communication system transports are not isolated from each other at CTT
- GPS has been incorporated into the CTT locomotives in the form of a "GPS Surrogate", which provides the GPS information that will come from the TMC in an actual PTC system deployment
- CTT does not have a separate hardwired Ethernet transport to its assets

CTTB is a test bed in the MCC lab that mimics the field setup at CTT as much as possible. From this perspective, CTTB is part of the CTT platform even though it has cabled RF connections and resides in a rack similar to the ILT SITBs. Along with its location, these are the only real differences between CTTB and the actual CTT field setup. All other facets are the same between CTTB and the CTT field setup.

To summarize, the CTT test setup consists of the following aspects:

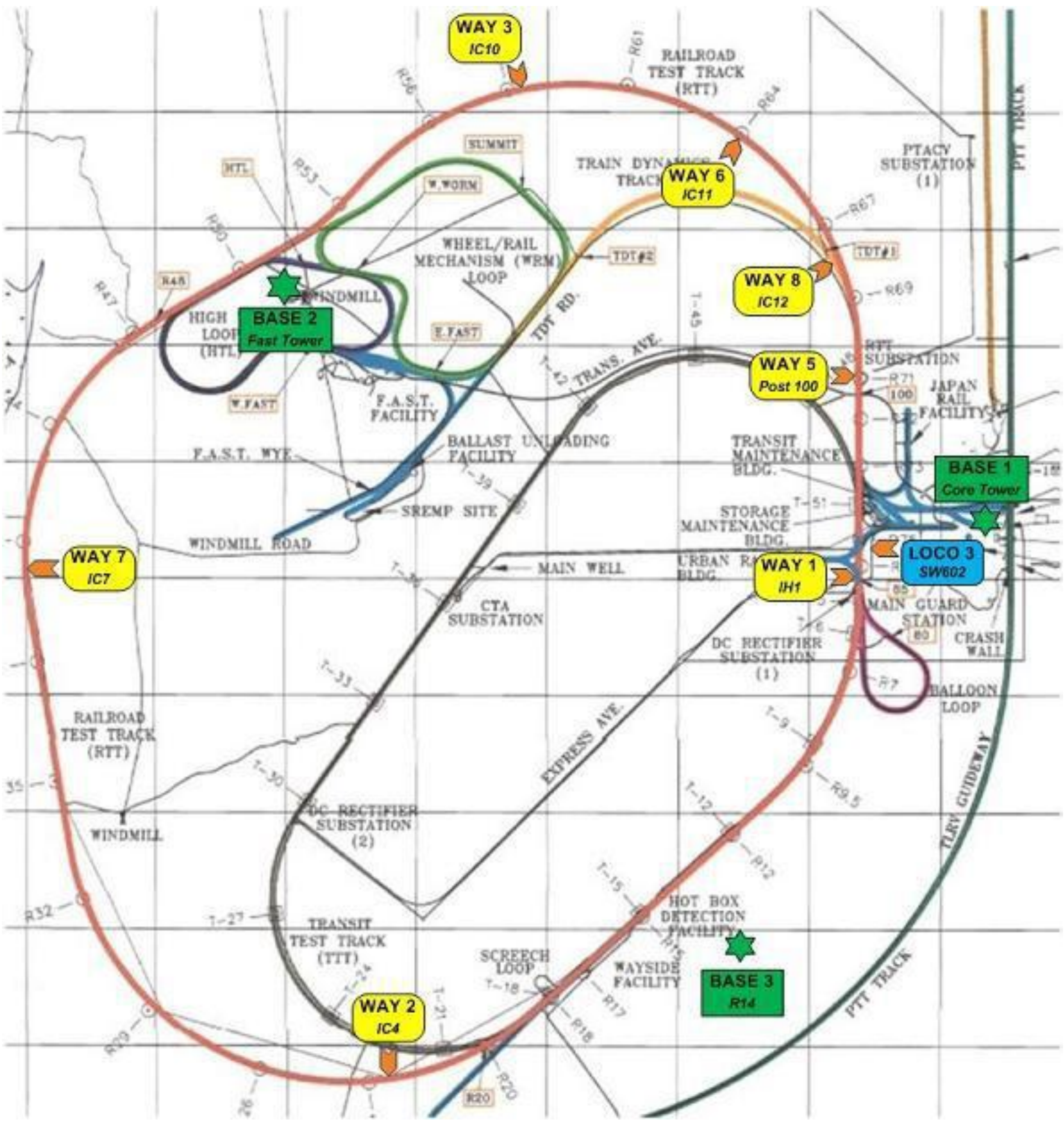
- Test-control connectivity to all assets via IP paths, typically OTA
- PTC communication sites including MCC's 220 MHz PTC radios:
    - 3 base stations
    - 3 locomotives
        - 2 located in actual GP40 locomotives
        - 1 stationary locomotive housed in an RTT bungalow
    - 7 waysides
- Alternate IP messaging paths (Cell and Wi-Fi) for all remotes
- On-site local BO servers hosting ITCM, ITCSM, and test-control SW (such as TE and the BO EPS SW)
- An SBC at each remote communication site that hosts the remote's ITCM SW (which functions as the remote ITCM server) and associated ITCSM agent
- An SBC at each remote communication sites that hosts the remote's EPS SW (which functions as either the WIU or TMC endpoint for a wayside or locomotive site, respectively)
- VPN connection to MCC in Renton, Washington that supports:
    - Connections to off-site BO(s) that can function as foreign, failover, and/or remote BO(s)
    - Connection to remote test control and results data-storage for analysis

Figure 14 shows the installation locations of the base station, wayside, and stationary locomotive communication sites at the CTT facility. The following notes these as well as the locations of other MCC assets:

- The 3 base stations are noted by green labels/stars and are identified as "BASE *x*".
- The 7 wayside sites noted in yellow and are identified as "WAY *x*". (Note that there is no 'WAY 4' site.)
- The stationary locomotive site is noted in blue and identified as LOCO 3.

- Not shown are the mobile locomotive sites or the BO:
  - When not moving on the RTT, LOCO 1 and LOCO 2, which are located in actual GP40 locomotives, are housed in or just outside of the locomotive "barn", which is just northwest of BASE 1 in Figure 14 (approximately one-quarter of the way towards WAY 5).
  - The on-site BO and Test-Control Center is located near the locomotive barn noted above.

Figure 14: CTT asset installation locations at TTCI



4.4.3 PTC communication sites at CTT

This section further describes the PTC communication sites at CTT. Table 15 shows the setup of each type of CTT site, which includes waysides, locomotives (both actual mobile and pseudo stationary sites), base stations as well as the BO and test-control sites.

Table 15 lists the salient components of each site where items like cabling, adapters, power supplies/converters are assumed. Detailed connection diagrams along with bills of material for each installation can be found in reference documents [3] and [7].

Table 15: PTC communication sites

Component	Wayside	Locomotive (Mobile)	Locomotive (Stationary)	Base Station	BO
Electrical Power	$120V_{AC}$	$74V_{DC}$ or $120V_{AC}$[i]	$120V_{AC}$	$120V_{AC}$	$120V_{AC}$
Climate Controlled	No	Only available when powered	No	Only 2 of 3 sites	Yes
PTC Radio	Yes	Yes		Yes	No
Cell Modem	Yes	Yes		No	No
Wi-Fi Modem	Yes	Yes		Yes[ii]	No
Wired Backhaul	n/a	n/a		Yes[ii]	n/a
ITCM and ITCSM SW Host	SBC	SBC		n/a	Server
EPS SW Host	SBC (WIU)	SBC (TMC)		n/a	Server (BO)
220 MHz Antenna	Yes	Yes		Yes	n/a
Cell Antenna	Yes	Yes		n/a	n/a
Wi-Fi Antenna	Yes	Yes		n/a	n/a
GPS Antenna	Yes	Yes		Yes	Yes
GPS Surrogate	No	Yes		No	No
Ethernet Switch	Layer-II	Layer-II		Layer-III [ii]	Layer-III

i. When the GP40 locomotive is powered on, it can supply $74V_{DC}$. $120V_{AC}$ is available when the GP40 is powered down but in proximity of a power outlet (that is, on "shore" power). The MCC locomotive PTC communication site is equipped to handle either source.

ii. Not all TTCI base stations originally had hardwire backhaul, which forced the use of Wi-Fi backhaul to the BO from those affected sites. However, all base stations now have hardwired backhaul and as such, their Wi-Fi modems are not currently in use.

4.4.3.1 *Waysides*

Figure 15 shows a typical wayside site at CTT. The antenna pole and nearby bungalow housing the PTC equipment are of particular interest. The heights of the wayside 220 MHz antennae at CTT range from approximately 12' to 30' high. The other system antennae (Cell, Wi-Fi, GPS) are generally placed on the bungalow roof or on the pole if space and logistics permit.

Figure 15: Typical CTT wayside site

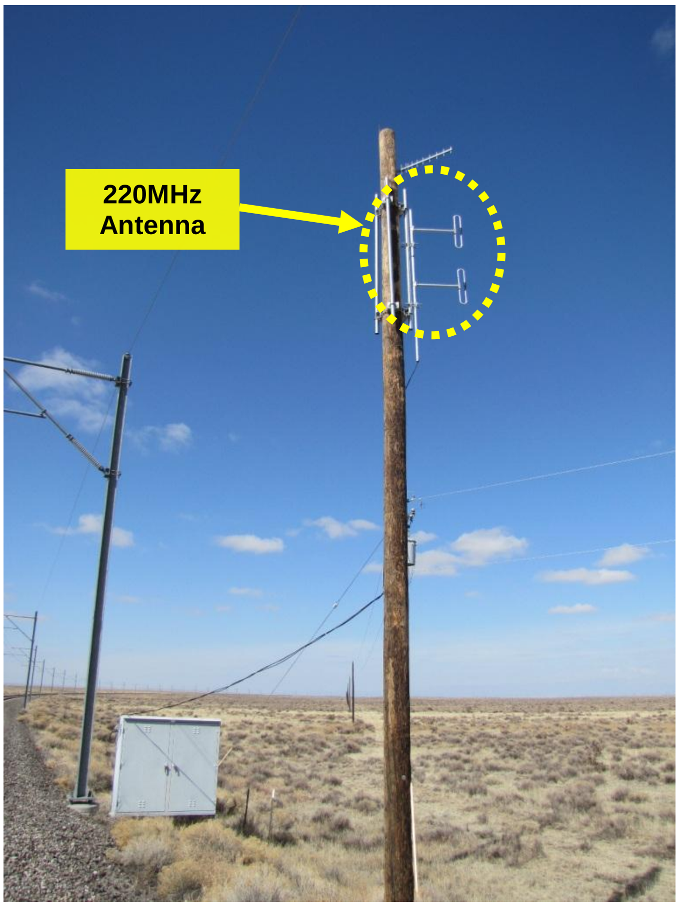
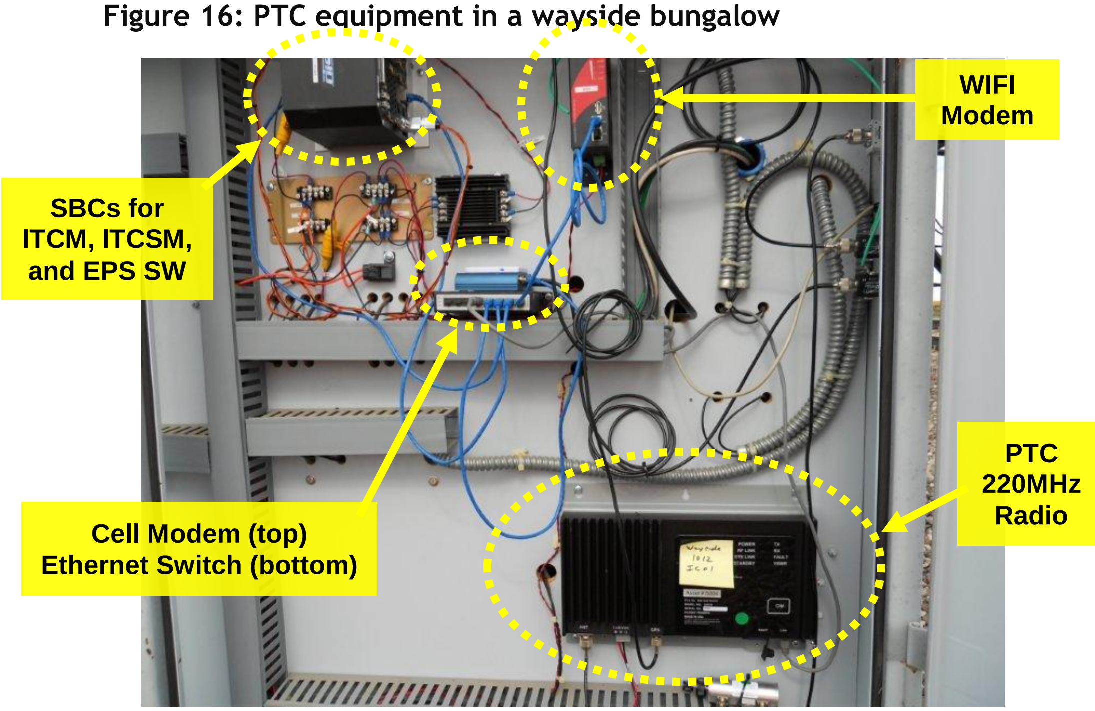


Figure 16 shows the interior of a typical wayside bungalow during installation at CTT. Note the PTC 220 MHz Wayside radio, dual-SBC package hosting pertinent SW as well as the IP equipment.

Figure 16: PTC equipment in a wayside bungalow



4.4.3.2 *Locomotives*

Figure 17 shows one of the two GP40 locomotives employed for CTT activities. The antenna platform is located near the front cabin on the roof of the locomotive.

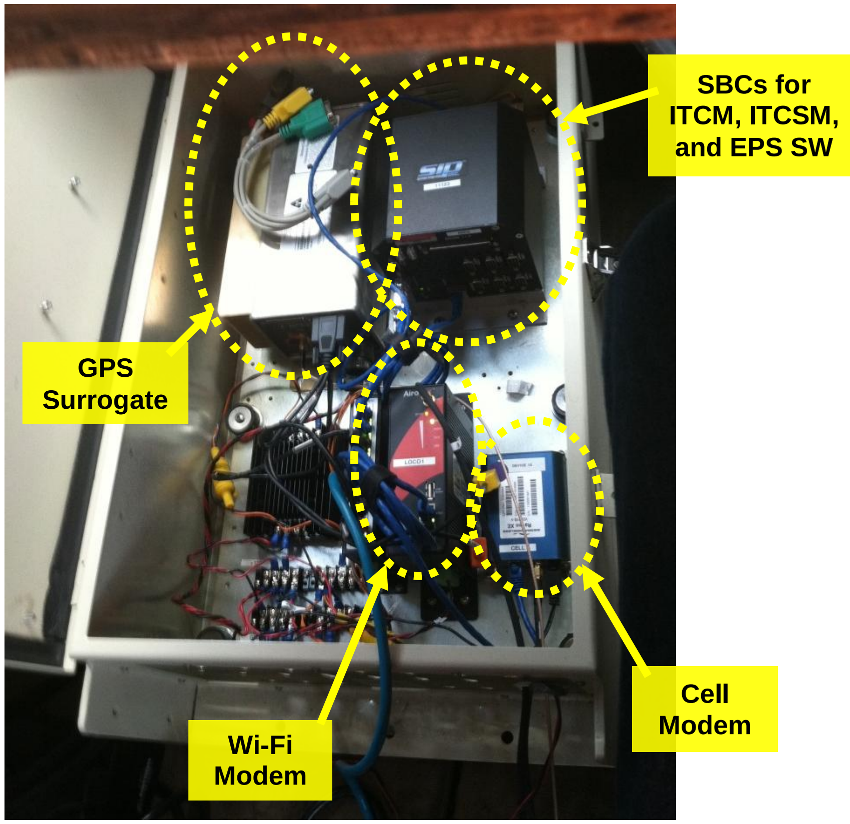
Figure 17: CTT GP40 locomotive

Figure 18 shows the 220 MHz antenna install on this platform. The locomotives' 220 MHz antenna height is approximately 18' to 19' high as measured to the track. The other system antennae (Cell, Wi-Fi, GPS) are placed further toward the front of the locomotive on the roof.

Figure 18: Typical locomotive 220 MHz antenna install

Figure 19 shows the interior of a typical locomotive casing in use at CTT. This casing holds all active equipment with the exception of the PTC 220 MHz Locomotive radio, which is made for mounting in the locomotive's LSI-standard rack. Similar equipment can be seen here as was shown in the previous subsection detailing the wayside setup. The primary exception here is the use of a GPS surrogate, which is a component that provides GPS packets over Ethernet and is unique to the CTT locomotive sites (whose radios do not have an onboard GPS Rx and instead will rely on the TMC for this information in a live, fielded PTC system setup).

Figure 19: Typical locomotive PTC equipment casing

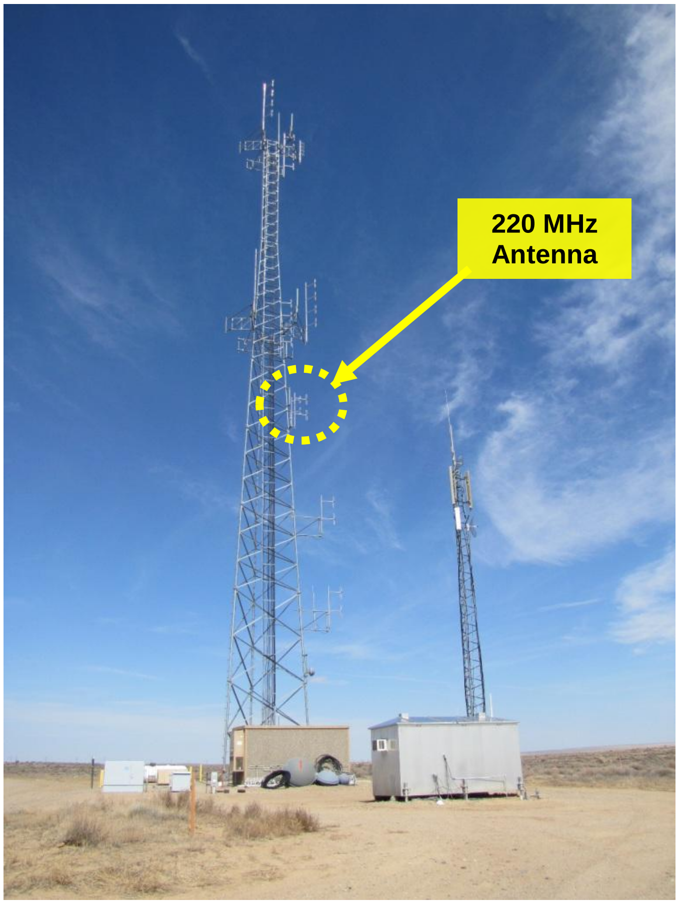


In addition to the two equipped GP40 locomotives, the CTT setup also has one “pseudo” locomotive site. This stationary site allows additional locomotive PTC message traffic to be generated as well as the baseline testing of a PTC locomotive communication site with no movement and additional locomotive noise. The stationary locomotive PTC equipment is housed at a vacant wayside site and as such, has a similar antenna setup to the waysides described in the previous section.

4.4.3.3 Base stations

Figure 20 shows a typical base station at CTT. Base stations are similar to the wayside infrastructure except they are usually much larger (for example, antenna towers instead of poles and housings with more space). Also, the CTT base stations are not necessarily located along the RTT like the wayside sites are. The heights of the base station 220 MHz antennae at CTT range from approximately 60’ to 110’ high, which are heights primarily resulting from available space on the respective towers.

Figure 20: Typical CTT base station



The amount of actual PTC equipment located at each base station is typically less than a remote site since much of the equipment associated with the base actually resides in the BO. As a result, the backhaul connection between the base station and BO is notable and has either been hardwired Ethernet (where and when available) or via Wi-Fi. Now that all TTCI base stations are physically wired, which offers more reliability, Wi-Fi backhaul has become a backup to be used only if and when needed.

4.4.3.4 BO and test control

The CTT BO site largely consists of various types of software and their corresponding hardware hosts. This includes the two RHEL-based servers housing the BO instances of ITCM, ITCSM, and EPS SW. It also includes the TE Server, which houses the primary engine behind most of CT's test control and processes many of the applications and data acquisition tools throughout the CTT setup. Figure 21 shows the local CTT control center.

Figure 21: CT's BO and test control

4.4.4 Remote test control at CTT

MCC has remote access to all PTC communication system test bed assets at CTT via the MCC-TTCI VPN connection. This includes BO systems, locomotive, wayside and base station sites. Remote test-control allows static testing (with stationary locomotives) at CTT that can be performed in a similar way to the testing on the lab-based SITBs.

As a result, through successive product release cycles, the amount of CTT executed remotely from MCC has greatly increased. This is largely due to the fact that needed resources are more prevalent and quickly available for any issues that may arise such as defecting potential system-level bugs. This development requires on-site support at CTT from TTCI personnel, which in addition to providing help to MCC with on-site scheduling of locomotive crews and resolving infrastructure issues, also includes assistance with PTC equipment updates and SW installations.

4.5 Test bed IP topology

Figure 22 shows the IP connections between the SITB assets as well as the ITCM transport paths for the ILT setup. Along with a 220 MHz PTC radio, each remote site has an SBC to host ITCM SW and the ITCSM agent, and Cell and Wi-Fi modems. Of particular note is the "Maintenance VLAN", which offers a physically isolated path for test-control communication to each asset than ITCM communication to each asset.

Figure 22: SITB asset connectivity

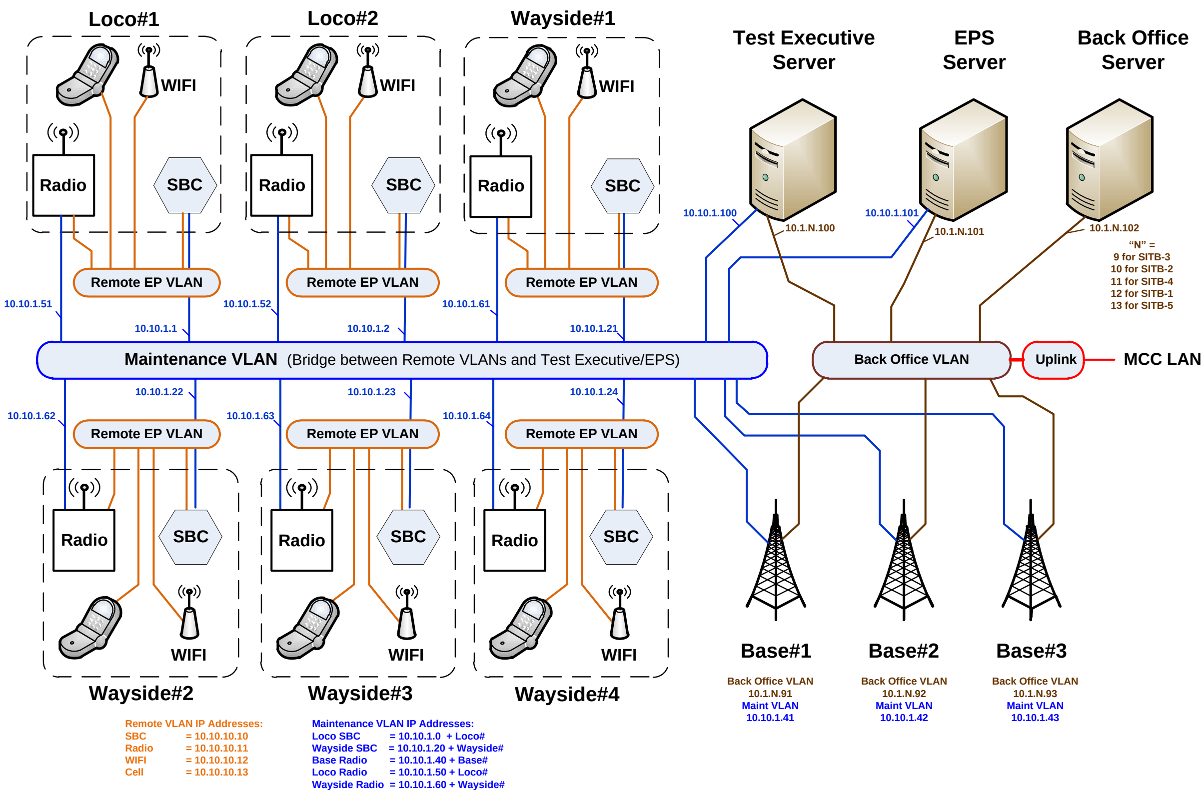
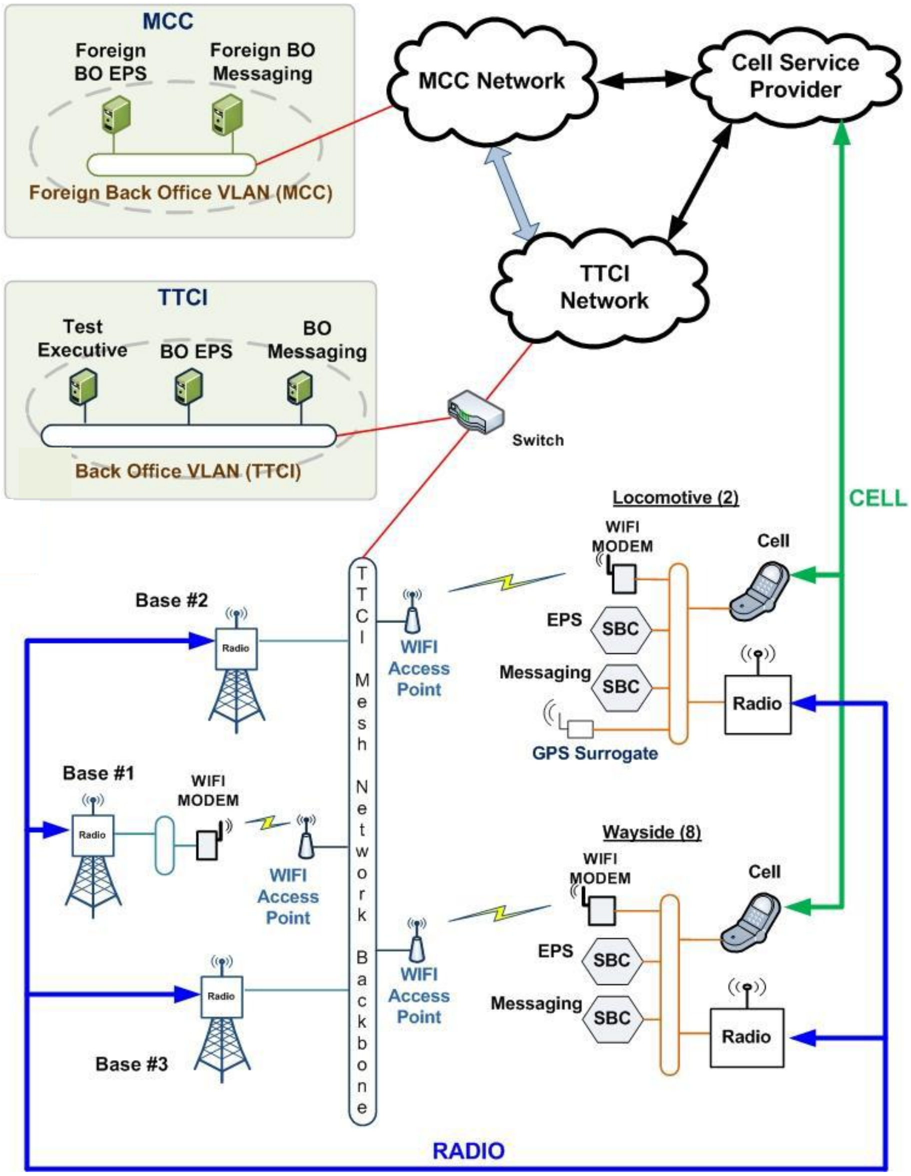


Figure 23 shows the IP connections between the assets as well as the ITCM transport paths for the CTT setup. Site-by-site, it is similar to the ILT setup except for many of the differences noted in Section 4.4.2. Of particular note is the use of an additional SBC in each remote for its EPS as well as the fact that test-control and ITCM routes share the same physical path to each asset.

DCN 00002483-A

Figure 23: CTT asset connectivity



5. System integration test control and data acquisition

Along with the test bed infrastructure and the PTC communication system itself, another fundamental piece of the IWS test activities is the IWS test-control system. Core to the test-control system are the several SW tools that facilitate test execution, data collection and analysis, subsystem and asset configuration, environmental manipulation as well as the other items highlighted in Section 4.3.5. These test SW tools consist of TE, the various sites' EPS SW, radio control applications, a results database, and scripts for everything from configuration and testing to data results parsing.

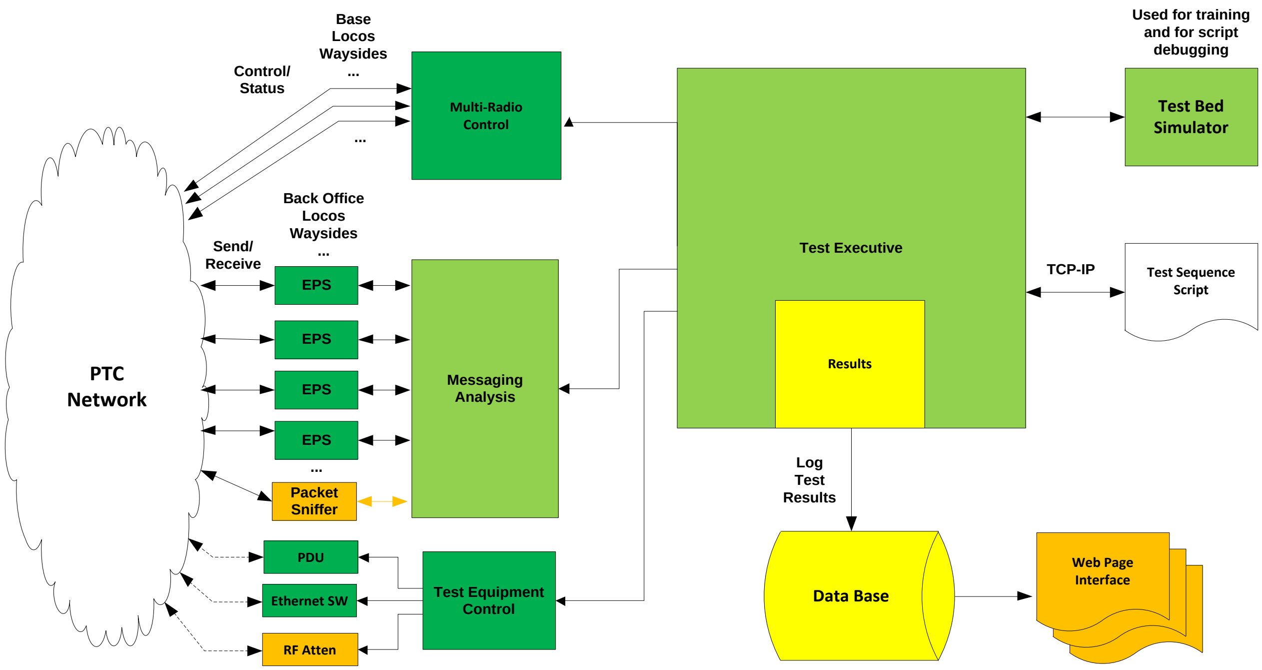
Over time through subsequent product release cycles, test control is expanded to accommodate increased system functionality and to increase overall test efficiency. An IWS test "framework" has been instituted with this in mind, which along with the IWS test tools, is described in this section.

5.1 Test-control SW tools

At a basic level, the IWS test-control system's SW tools operate each IWS test bed, whether the SITBs in ILT or the assets at CTT. Figure 24 shows many of the prominent IWS SW tools and how they interconnect with each other and into the PTC communication system.

DCN 00002483-A

Figure 24: Integration test-control SW tools and their interconnects



5.1.1 Test executive

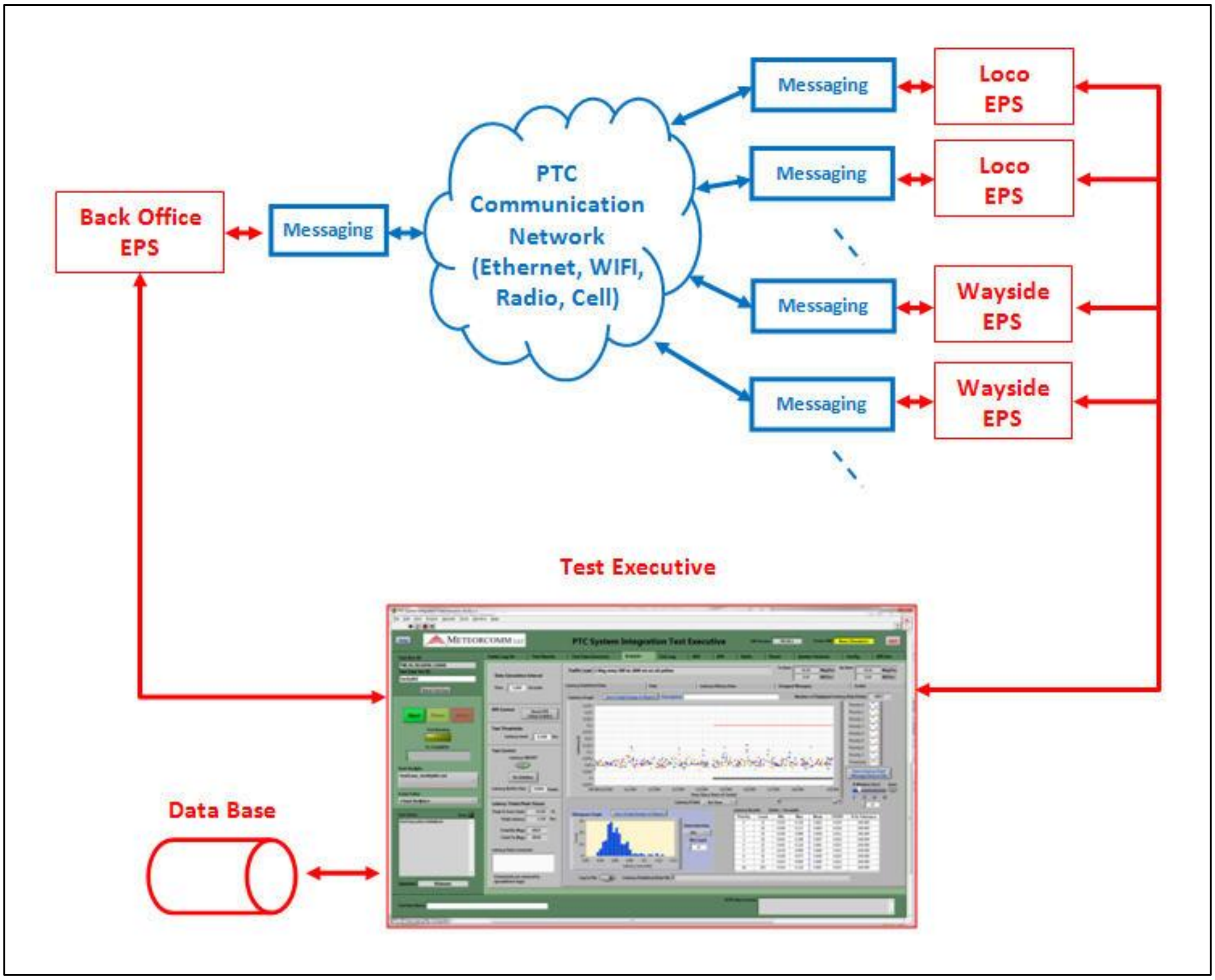
At the heart of the IWS test-control system is the Test Executive (TE). TE is both a stand-alone application as well as an environment where many of the IWS SW tools can be operated and controlled from a single point. Each "sub-application" is typically designed to be used as a stand-alone application as well as controlled by TE or one of its components. TE is the top-level integration test SW application and consists of 3 major components, which comprise the "brains" of the IWS test-control system. These include the following:

- TE application: This application resides on a Windows-based server and functions as the primary user interface allowing the operator to perform manual or automated tests and data-analysis activities.
- EPS(s): This application serves in place of each communication site's endpoint as the interface between TE and the PTC communication system. EPS sends messages into the system based upon commands received from TE, and returns message-traffic information and status back to TE for analysis.

- Database: Test results captured by TE are stored as data log-files that can be analyzed later using reporting tools.

Figure 25 shows these major components of TE and also indicates how they embrace the test bed and then generate and consume messages in order to collect data. (Note that although Figure 25 shows the EPS SW interfacing with ITCM, it could also just as well be interfacing with ITCSM.)

Figure 25: The "brains" of the integration test-control system



5.1.1.1 *Manual and automated (script) control*

TE has a rich user interface that makes it possible to perform elaborate manual tests. Manual tests can generate reports and create log-files of data. The user can select a variety of message types to be sent over various communication paths. The messages can be set up to be sent indefinitely or until a certain number of messages have been sent as well as set up to be sent at a specific rate or randomly. Any applicable level of QoS parameter as well as message sizes can be varied as needed.

Similarly, test scripts or a controlling application may be used to command TE in an automated fashion and execute any feature that could otherwise be operated manually. In this case, the script or application serves as the master (instead of a tester) and the TE is the slave. The script or application sends commands to the TE that either induce a NAK response or an ACK response plus data.

TE facilitates automation by acting as a TCP-IP server and listening for a controlling application or script to direct it. Commands and status are passed over this TCP-IP interface, which is intended to be used to perform a sequence of steps to run automated test cases.

With regard to test-script control of TE, the following guidelines should be noted:

- Scripts can be in any language that supports a TCP-IP connection as long as they follow the TE API (in order to interact with TE)
- Scripts drive the commands that TE will send to the EPSs
- Scripts define the evaluation criteria that TE uses (for example, for determining PASS/FAIL)

5.1.1.2 Data metrics

TE has been developed to analyze the following criteria in assessing PTC communication system functionality and performance levels:

- Message Success Rates
- Message Latency
- Message Path Selection
- Message Quality of Service (QoS)
- Data Throughput
- Traffic Load
- Fault Recovery

5.1.1.3 Database

A SQL database has been developed to collect and store integrated system test results. During the test process, TE connects to this database and sends log-files as inputs to the database. Information is accessed by any tool capable of connecting to a database and running SQL queries. As such, the TE database has reporting tools to analyze test results, which also includes a web interface.

5.1.1.4 *Infrastructure control*

As a component of TE, the Test-Equipment Control tool interfaces to multiple items of electronically controlled test equipment. Specifically, this application controls the following electronically controlled components in real-time by sending commands and receiving and processing their status:

- Power Distribution Unit (PDU): Controls the power supplies of all active components at each individual communication site
- Ethernet Switch: Controls each transport's physical connection at each individual communication site
- RF variable attenuators: Controls the 220 MHz mobile-radio-to-other-radio RF path loss

As introduced in Section 4.3, the first two electronically controlled components above apply only to ILT. It should be noted that the capability of this tool can be expanded to more test equipment items if needed.

5.1.2 End point simulator

The End Point Simulator (EPS) is an application that simulates an endpoint application running at any of the PTC communication sites, which are noted as follows:

- a BO application running in the BO
- a WIU application running at a wayside site
- a TMC application running at a locomotive site

As such, all BO, wayside, and locomotive sites throughout IWS require an EPS. During actual system integration testing, multiple instances of the EPS application will be running in parallel to support PTC communication system functionality and data collection at all communication sites that are included in a given test case.

The EPS interfaces with TE on one side and the PTC communication system SW on the other. At the TE-EPS interface, TE will send commands to the EPS, which in turn will send messages into the system onto either ITCM or ITCSM as applicable. Collectively, the EPSs at both the source and destination assets notify TE of sent and received PTC message information. TE uses this as well as other EPS-provided information (such as timestamps) to perform analysis.

On the PTC communication SW side, EPS interfaces with both ITCM and ITCSM, both of which reside on an SBC at the remote sites as well as servers in the BO. As such, EPS supports EMP and ISMP payload protocols as well as AMQP and Class D delivery protocols. EPS facilitates PTC communication SW functions as follows:

- ITCM: The EPS generates and receives message traffic.
- ITCSM: The EPS executes SMS sequences, which includes verification of proper agent responses.

The EPS can be controlled by any application that connects to it over TCP-IP and follows its API. Typically, its controlling application is a component within TE, which it receives commands from and sends gathered information to. EPS runs on either a 32 or 64-bit computer with the RHEL OS and as a result, either resides on a BO server or on an SBC at the remote site (for CTT only).

For illustration, the TE-EPS-PTC communication system interaction is outlined as follows:

1. Test Executive sends commands to EPS(s).
2. EPS(s) generate application-level messages that go into the PTC communication system:
   - In ITCM, this will be basic messaging.
   - In ITCSM, this will be specific SMS activities such as a kit distribution.
3. EPSs measure attributes related to the above PTC communication system operations and reports that information back to TE.
4. TE evaluates performance, logs results, and stores data.

5.1.3 Radio control

The following subsections describe the two applications that have been used for monitoring the PTC 220 MHz radios during IWS testing.

5.1.3.1 *Multi-radio control application*

The Multi-Radio Control application interfaces to multiple PTC 220 MHz radios. This application interrogates the radio(s) at a desired cadence and builds data tables and graphs presenting commonly-needed radio information to the user and the data log-files. This commonly-needed information includes the following:

- Radio run-time status
- Radio IDs
- Radio SW version
- Radio status and self-test results:
  - Number of resets
  - Host mode
  - RF link information such as Tx status and RSSI levels
  - GPS status
  - DSP mode
- Message traffic across the 220 transport and ITCR network on a radio-by-radio basis

Additionally, similar to the XTermW application described in the next subsection, the Multi-Radio Control application also supports the use of a terminal window (telnet) for each radio where a user can communicate with the radio in real-time. The application also has features including data reporting, highlighting, SW download, script file execution, and (user-definable) macro functionality.

The Multi Radio Control application runs on a Windows-based server and can be launched either as a component within TE or as a stand-alone application. In stand-alone mode, it listens on a TCP-IP port for a controlling application or script to control it.

5.1.3.2 *XTermW*

XTermW is a legacy terminal emulation application that runs on a Windows OS and enables a user to connect to a single radio for real-time interaction. Multiple instances of the application are operated simultaneously for interfacing to multiple radios.

The functionality of XTermW is very similar to the Multi-Radio Control application's telnet functionality noted in the previous subsection. It has features allowing radio SW downloads, script file editing and execution, user-defined macros, and highlighting, which is very useful for filtering out events in a high-density radio-trace output.

5.1.3.3 *SocketShare*

SocketShare is an Ethernet port-sharing utility that runs on a Windows OS. It allows multiple connections to a single Ethernet port and will also log any outputs from that single Ethernet port. SocketShare is used as a means to allow multiple applications to connect to each radio's Ethernet port, which is typically set to stream event and data traces in real-time.

5.2 Test-control framework

As mentioned earlier in the document, test automation is essential to IWS test activities; not only to improve test efficiency but to fundamentally allow test coverage of the regularly growing amount of system requirements and overall system functionality. IWS's test-tool and script development activities enable test automation. In tandem with these development activities, an IWS test-control framework has been put into place, which establishes the necessary test processes and structure to support integrated test automation. These test processes and structure are further described as follows:

- The test processes help define how the test tools and scripts interact with each other as well as with the test bed infrastructure and PTC system components. The development of these processes solidify the roles of existing test tools and identify holes where other test tools are needed – both of which help identify and scope the needed functionality of each test tool.
- The structure helps assure that the latest and most applicable version of each piece of test control SW – from configuration to test execution scripts – is being used and is easily accessible across all test beds by any user.

5.2.1 Test scripts

At the core of the IWS test framework and automation capabilities are the numerous test scripts used for test execution and configuration. IWS scripts have been primarily written in three languages as follows:

- Linux shell scripts: For conducting basic actions.

- Python scripts: For conducting more in-depth functions where use of libraries is necessary.
- TE's scripting language: Legacy language used for test-execution scripting to support early IWS test-automation efforts. This language is being retired in lieu of Python use.

Test execution scripts are usually telling TE what system activities to conduct during a particular test case (for example, the types of messaging and what assets to include). These scripts have been written in all of the various languages noted above. As noted in Section 5.1.1.1, test execution scripts include pass/fail criteria and per-test-case asset setup.

Test configuration scripts include test bed and PTC communication system setup, typically applicable to several test scenarios and/or entire test sprints. They are commonly written as Python or Linux shell scripts but also include subsystem-specific scripting languages. Example tasks of test configuration scripts include the following:

- Kick-starting server/SBC RHEL OS installation
- Deployment of ITCM SW and its topology files
- Configuring test bed radios

5.2.2 Subversion (SVN)

Control of the configurable elements of the test infrastructure and PTC communication system components (such as configuration files, test SW, and test scripts) is managed through SVN, an open-source version control tool.

5.2.3 HPQC

HPQC is the IWS test management tool and is detailed throughout Section 3.4. In a nutshell, HPQC stores test cases and requirements (and maps between them), tracks test progress, stores test results, and manages defects through resolution.

5.3 Typical test case

A typical test case using the IWS test tools and framework operates as follows:

- Framework accesses the applicable test case(s) in HPQC.
- The appropriate test scripts are invoked, which open a socket with TE and passes it commands.
- TE sets up the test, starts a log-file, and sends commands to the applicable EPS instances for the BO and remote(s)
- EPS generates the various messages according to TE instructions and send messages and commands to the ITCM asset at each communication site involved in the test:
    - The various ITCM assets send the messages over the appropriate communication transport(s) (for example, 220, Cell, Wi-Fi, or hardwire).
- Based on the specific test case, messages are received by the ITCM assets at the appropriate destinations:
    - The various destination ITCM assets provide the messages to their respective EPS instances for consumption.
- EPS sends its messaging data to TE, which it uses to determine latency and other results, then logs, and finally sends onto the test script.
- The test script consumes its desired information from TE, decides if the test case PASSED or FAILED, and then sends that data to TE and Framework.
- Framework updates the test results in HPQC.

Appendix - Test report plots

The following plots are examples of the information contained in the IWS Weekly Reports, which is discussed in Section 3.4.2.3.

Figure 26 shows the cumulative number of test cases within a test sprint that were planned for and actually tested during a week. The initial quantity is the amount of test cases that had been tested as of the end of the previous week.

Figure 26: Example integration test case burn-through

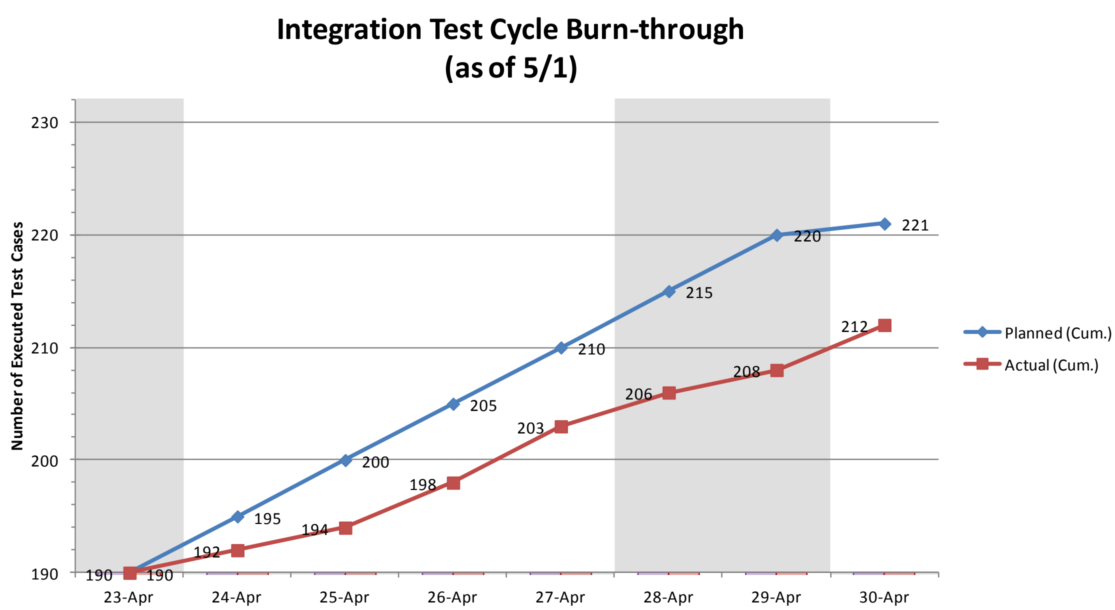
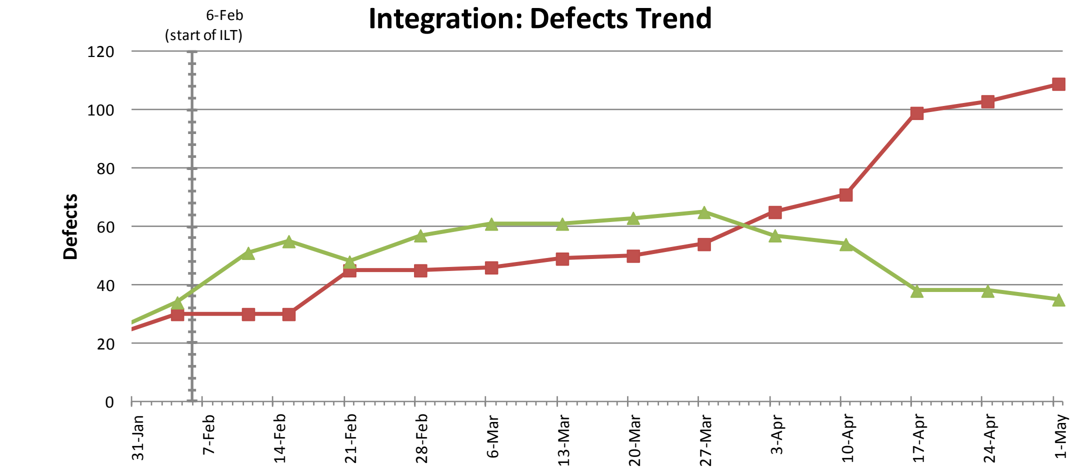


Figure 27 shows the trends of product defects being found through integration testing. The two traces shown are the quantity of defects being CLOSED versus the quantity of defects that are “In Progress”, which includes defects that are NEW, OPEN or with a FIX that are still waiting validation. The trend lines give a quick view of how many defects have been found during the test cycle and how they are being addressed.

Figure 27: Example trend of defects found during integration testing



	28-Jan	4-Feb	11-Feb	15-Feb	21-Feb	28-Feb	6-Mar	13-Mar	20-Mar	27-Mar	3-Apr	10-Apr	17-Apr	24-Apr	1-May
Closed	22	30	30	30	45	45	46	49	50	54	65	71	99	103	109
In Progress	23	34	51	55	48	57	61	61	63	65	57	54	38	38	35

Figure 28 gives a breakdown of the priority and status of the product defects found during integration testing. This plot gives a quick view of the number of higher and lower priority defects working through the MCC defect resolution process.

Figure 28: Example snapshot of integration-found defects by status