CMS Manual System	Department of Health & Human Services (DHHS)
Pub 100-01 Medicare General Information, Eligibility, and Entitlement	Centers for Medicare & Medicaid Services (CMS)
Transmittal 97	**Date: January 15, 2016**
	Change Request 9219

Transmittal 95, dated October 16, 2015, is being rescinded and replaced by Transmittal 97, dated January 15, 2016, to restore language in Section 40.1, and replace the Standard Terminology chart in section 40.1.1. All other information remains the same.

SUBJECT: Internet Only Manual (IOM) Publication 100-01 - General Information, Eligibility, and Entitlement, Chapter 7 - Contract Administrative Requirements, Section 40 – Shared System Maintainer Responsibilities for Systems Releases

I. SUMMARY OF CHANGES: Updating information found in IOM Publication 100-01, Chapter 7, Section 40 and its subsections. General Information, Eligibility and Entitlement Contract Administrative Requirements - Shared System Maintainer Responsibilities for System Releases

Medicare Administrative Contractor (MAC) updates to Chapter 7, Section 40 and its subsections will be updated with a future Change Request.

EFFECTIVE DATE: September 21, 2015
**Unless otherwise specified, the effective date is the date of service.*
IMPLEMENTATION DATE September 21, 2015
Disclaimer for manual changes only: The revision date and transmittal number apply only to red italicized material. Any other material was previously published and remains unchanged. However, if this revision contains a table of contents, you will receive the new/revised information only, and not the entire table of contents.

II. CHANGES IN MANUAL INSTRUCTIONS: (N/A if manual is not updated)
R=REVISED, N=NEW, D=DELETED-*Only One Per Row.*

R/N/D	CHAPTER / SECTION / SUBSECTION / TITLE
R	Table of Contents
R	7/40/Shared System Maintainer and Part A/Part B (A/B)/Durable Medical Equipment (DME) Medicare Administrative Contractor (MAC) and the Single Testing Contractor (STC) Responsibilities for Systems Releases
R	7/40.1/Standardized Terminology for Claims Processing Systems
R	7/40.1.1/Standard Terminology Chart
R	7/40.2/Release Software
R	7/40.2.1/ Implementing Validated Workarounds for Shared System Claims Processing by All Medicare DME MACs
R	7/40.3/Shared System Testing Requirements for Shared System Maintainers, Single Testing Contractor (STC)/Beta Testers, and Part A/Part B (A/B) Durable Medical Equipment (DME) Medicare Administrative Contractors (MACs)
R	7/40.3.1/Shared System Testing Requirements for Shared System Maintainers, Single Testing Contractor (STC), and DME MACs
R	7/40.3.2/Minimum Testing Standards for Shared System Maintainers and the Single Testing Contractor (STC)/Beta Testers
R	7/40.3.3/Testing Standards Applicable to all Beta Testers
R	7/40.3.5/Part A/Part B (A/B) Durable Medical Equipment (DME) Medicare Administrative Contractor (MAC) (User) Testing Requirements
R	7/40.3.6/Testing Requirements Applicable to all CWF Data Centers (Hosts)
R	7/40.3.7/Timeframe Requirements for all Testing Entities
R	7/40.3.8/Testing Documentation Requirements
R	7/40.3.9/Definitions
R	7/40.3.10/Test Case Specification Standard
R	7/40.3.11/Next Generation Desktop (NGD) Requirements

III. FUNDING:
For Medicare Administrative Contractors (MACs):
The Medicare Administrative Contractor is hereby advised that this constitutes technical direction as defined in your contract. CMS does not construe this as a change to the MAC Statement of Work. The contractor is not obligated to incur costs in excess of the amounts allotted in your contract unless and until specifically authorized by the Contracting Officer. If the contractor considers anything provided, as described above, to be outside the current scope of work, the contractor shall withhold performance on the part(s) in question and immediately notify the Contracting Officer, in writing or by e-mail, and request formal directions regarding continued performance requirements.

IV. ATTACHMENTS:
Business Requirements
Manual Instruction

Attachment - Business Requirements

Pub. 100-01	Transmittal: 97	Date: January 15, 2016	Change Request: 9219

Transmittal 95, dated October 16, 2015, is being rescinded and replaced by Transmittal 97, dated January 15, 2016, to restore language in Section 40.1, and replace the Standard Terminology chart in section 40.1.1. All other information remains the same.

SUBJECT: Internet Only Manual (IOM) Publication 100-01 - General Information, Eligibility, and Entitlement, Chapter 7 - Contract Administrative Requirements, Section 40 – Shared System Maintainer Responsibilities for Systems Releases

EFFECTIVE DATE: September 21, 2015
**Unless otherwise specified, the effective date is the date of service.*
IMPLEMENTATION DATE: September 21, 2015

I. GENERAL INFORMATION

A. Background: Internet Only Manual (IOM) Publication 100-01 General Information, Eligibility, and Entitlement, Chapter 7 - Contract Administrative Requirements, Section 40 – Shared System Maintainer Responsibilities for Systems Releases

B. Policy: N/A

II. BUSINESS REQUIREMENTS TABLE

"Shall" denotes a mandatory requirement, and "should" denotes an optional requirement.

Number	Requirement	Responsibility								
		A/B MAC			DME MAC	Shared-System Maintainers				Other
		A	B	HHH		FISS	MCS	VMS	CWF	
9219.1	Shared System Maintainers shall follow instructions outlined in attached IOM publication 100-01, Chapter 7 updates.					X	X	X	X	
9219.2	The Single Testing Contractor (STC) shall follow instructions outlined in attached IOM publication 100-01, Chapter 7 updates.									STC
9219.3	When communicating with CMS, the Shared System Maintainers (SSMs) and the Single Testing Contractor (STC) shall use the standard terminology in the standard terminology chart in section 40.1.1. SSMs and the STC may continue to use their own unique terminology for their internal processes. Using					X	X	X	X	STC

Number	Requirement	Responsibility								
		A/B MAC			DME MAC	Shared-System Maintainers				Other
		A	B	HHH		FISS	MCS	VMS	CWF	
	standard terminology will minimize confusion and misunderstanding for CMS, SSMs, and the STC while participating in discussions.									

III. PROVIDER EDUCATION TABLE

Number	Requirement	Responsibility				
		A/B MAC			DME MAC	CEDI
		A	B	HHH		
	None					

IV. SUPPORTING INFORMATION

Section A: Recommendations and supporting information associated with listed requirements:

"Should" denotes a recommendation.

X-Ref Requirement Number	Recommendations or other supporting information:

Section B: All other recommendations and supporting information: N/A

V. CONTACTS

Pre-Implementation Contact(s): Barb Pecoraro, 410.786.6188 or barbara.pecoraro@cms.hhs.gov

Post-Implementation Contact(s): Contact your Contracting Officer's Representative (COR).

VI. FUNDING

Section A: For Medicare Administrative Contractors (MACs):
The Medicare Administrative Contractor is hereby advised that this constitutes technical direction as defined in your contract. CMS does not construe this as a change to the MAC Statement of Work. The contractor is not obligated to incur costs in excess of the amounts allotted in your contract unless and until specifically authorized by the Contracting Officer. If the contractor considers anything provided, as described above, to be outside the current scope of work, the contractor shall withhold performance on the part(s) in question and immediately notify the Contracting Officer, in writing or by e-mail, and request formal directions regarding continued performance requirements.

General Information, Eligibility, and Entitlement Manual

Chapter 7 - Contract Administrative Requirements

Table of Contents
(Rev.97, Issued: 01-15-16)

Transmittals for Chapter 7

40 – Shared System Maintainer, Part A/Part B (A/B)/Durable Medical Equipment (DME) Medicare Administrative Contractor (MAC) and the Single Testing Contractor (STC) Responsibilities for System Releases

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

40.1 – Standardized Terminology for Claims Processing Systems

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

This is a responsibility for all the Part A/B, DME (MACs) and the Single Testing Contractor (STC). Medicare requires implementation of a limited number of shared systems by all MACs for their claims process and related functions. This eliminates the need for each MAC to repeat development of the base system.

The shared system maintainers, MACs and the STC shall use standardized terminology to refer to common systems maintenance elements in all discussions, reporting, and documentation. A chart of topics, showing how each system currently refers to them, and what they are called is located at 40.1.1. The list is not exhaustive and both CMS and the shared system maintainers shall add to it, deciding with each addition, the common term we shall use to describe it. The MACs and the STC have a stake in this standardization, since many access the Information Management (INFOMAN) databases each shared system maintainer populates and maintains to determine the status of changes of interest to them. Using common terminology will minimize confusion and misunderstanding for CMS, SSMs, MACs and the STC while participating in discussions.

The MACs and the STC shall examine their use of the system status information issued by the Shared System Maintainers to determine if they have internal applications that need to be adjusted to adopt the standardized terminology. If they have internal systems or processes that must be modified to reflect the standardization required by this instruction, they shall make those changes to coincide with the shared system changes.

40.1.1 - Standard Terminology Chart

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

STANDARD TERM FOR ALL SHARED SYSTEM MAINTAINERS (SSM) AND THE SINGLE TESTING CONTRACTOR (STC)	**DESCRIPTION**
QUESTION	Request for assistance and/or reported potential system problem
PROBLEM	Confirmed system and/or documentation problems
CR	Change Request - Any software modification made to the system as a result of a CMS mandate, user or maintainer initiated action
CMS STATUS	CMS needs take action by answering a question, finalizing an instruction, etc.
CONF STATUS	PROBLEM, CR or proposed action is under discussion in a functional workgroup
NSC (non-system change) STATUS	CMS CR does not require shared system change. May require A/B MAC maintenance
RESEARCH STATUS	The system maintainer completes high level review of required changes by analyzing them and determining the intent of the change request
REQS STATUS	The system maintainer finalizes the business requirements

STANDARD TERM FOR ALL SHARED SYSTEM MAINTAINERS (SSM) AND THE SINGLE TESTING CONTRACTOR (STC)	DESCRIPTION
WALKTHROUGH STATUS	The system maintainer presents the systems solution to the CR in a structured walkthrough discussion with CMS and Beta testers
WORK STATUS	The system maintainer completes technical design, coding and unit testing the system change
ALPHA TESTING	Maintainer System Testing
BETA TESTING	Testing (Beta)
UAT	User Acceptance Testing
USER STATUS	Back to user to provide more information or examples, assess solution
SCHED STATUS	Scheduled to go out with a release date assigned for implementation
RESOLVED	PROBLEM has been resolved: question answered, potential system anomaly explained or correction identified and scheduled for release
RELEASE	Quarterly Releases are planned and scheduled in advance; normally contain changes due to routine maintenance
FOLLOW-UP	What Maintainers send out to augment a release or correct PROBLEMS directly related to a newly-installed release
EMER RELEASE	What Maintainers send out to fix Priority 1, 2, and urgent PROBLEMS or when there are critical processing errors that must be resolved immediately
OFF-QTR RELEASE	What Maintainers send out to fix non-urgent PROBLEMS between releases or to prepare for an upcoming release (e.g. update provider profile data)
TEST CASE	A description of an input situation and of the expected results associated with a specific test objective. (a Test Case may optionally include test steps provide to additional granularity)
TEST SET	A group of test cases with a common goal (e.g., a test set to validate a specific CR, a regression test set)

40.2 – Release Software

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

CMS intends to continue to closely manage shared system software changes to assure that an effective change control process is in place. This means that maintainers must receive approval from their CMS system maintenance lead/Government Task Leader (GTL) (see section VI) or CMS project officer before any follow-up release by the shared system maintainer can be scheduled and installed.

Control of System Changes

All maintainers of the shared systems (CWF, FISS, MCS and VMS, systems) must use the same quarterly release schedule, i.e., on or about January 1, April 1, July 1, and October 1. The specific schedule for each quarterly release will be determined by CMS.

All follow-up release changes (except emergencies) to the quarterly schedule must be held and released on a predetermined schedule in coordination with CMS. Unscheduled emergency changes released as problems are identified without prior approval. The schedule for a follow-up release of changes must be forwarded to your CMS system maintenance lead/(GTL) or CMS project officer for prior approval.

Follow-up release changes are to be limited to the correction of priority 1 and 2 problems and errors that prevent effective operation of the production system. Priority 3, priority 4 and/or priority 5 problems may be corrected in a follow-up release when pre-approved by CMS. The CMS maintenance lead/(GTL) will advise you of the approval decision within 24 – 48 hours.

When a system problem is identified, Medicare organizations (i.e. shared system maintainers, the Single Testing Contractor (STC), MACs and CWF Hosts) must submit documentation to their CMS system maintenance lead/(GTL) outlining the problem and the reason correction is needed at this time. Section V of this instruction outlines the minimum information required by CMS for approval.

Problem Priority Classifications for Follow-Up Releases

Listed below are CMS's problem priority classifications and examples.

Priority 1 Classification

Production:

The problem prevents the accomplishment of a mission critical capability for which no acceptable workaround is known.*

This priority also includes problems where code must be fixed immediately in order for the normal production region functions or services to continue. For example, if the production region is down in a job resulting in an incomplete cycle or the system is pricing a significant volume of claims incorrectly causing over or under payment. The maintainer may make priority 1 changes on its own authority. These corrections must be reported to the CMS maintenance lead/(GTL) or to the project officer the next business day.

EXAMPLES:

- ABENDS on-line or batch (Inability to run a cycle)
- Inaccurate payment or no payment of claims (significant impact/high volume)
- Necessary file updates cannot be accomplished (payment files, history files)
- Interface failures affecting claims processing

Beta/User Acceptance Testing:

The problem would prevent the accomplishment of a mission critical capability if the current test software is moved into the production environment. This priority also includes problems where code must be fixed immediately in order for the normal test region functions or services to continue. For example, if the test region is down in a job causing the cycle to not complete or the system is pricing claims incorrectly with a potentially significant claim volume or payment impact, the issue would be classified as a priority 1. The maintainer must work immediately to code a fix to be installed before moving the software into production.

EXAMPLES:

- ABENDS; inability to run a cycle or test
- Inaccurate payment or no payment of claims (potentially significant impact)
- Necessary file updates cannot be accomplished (payment files, history files)
- Interface failures affecting test conditions

Priority 2 Classification

Production:

The problem adversely affects the accomplishment of a mission critical capability so as to degrade performance and for which no acceptable work-around is known.* This means the problem adversely affects the payment of benefits with a small claim volume or payment impact, the completion of CMS required reporting, or inaccurate information is being sent providers, beneficiaries or CMS. For example, if the information on an outgoing document to the provider community or Medicare Summary Notice is incorrect, the issue would be classified as a priority 2. The system maintainer must work with the CMS maintenance lead/(GTL) for approval to implement a fix.

EXAMPLES:

- Inaccurate payment or no payment of claims (small impact/low volume)
- Inaccurate CMS required report
- Inaccurate messages to the beneficiary, provider or CMS
- ABENDs with limited impact (ex. One contractor)

Beta/User Acceptance Testing:

The problem would adversely affect the accomplishment of a mission critical capability so as to degrade performance if current test software is moved into the production environment. This means the problem adversely affects the payment of benefits with a potentially small claim volume or payment impact, the completion of CMS required reporting, or inaccurate information is being sent to providers, beneficiaries or CMS. For example, if the information on an outgoing document to the provider community is incorrect, the issue would be classified as a priority 2. The maintainer must work immediately to code a fix to be installed before moving the software into production.

EXAMPLES:

- Inaccurate payment or no payment of claims (potentially small impact)
- Inaccurate CMS required report
- Inaccurate messages to the beneficiary, provider or CMS

Priority 3 Classification

Production:

The problem adversely affects the accomplishment of mission critical capability so as to degrade performance and for which an acceptable workaround is known.*

This means the problem could have significant impact but the work-around alleviates the impact. This allows the system maintainer adequate time to code a fix and sufficiently test before the corrected software is delivered for production installation. The system maintainer must work with the CMS maintenance lead/(GTL) to implement a fix.

EXAMPLES:

- Impact of problem could be significant or minimal
- Problem correctable by contractor workaround*
- ABENDs with an acceptable workaround*

Beta/User Acceptance Testing:

The problem would adversely impact the accomplishment of a mission critical capability so as to degrade performance if current test software is moved into the production environment.

If moved into the production environment before correcting an acceptable workaround could be instituted to prevent the adverse impact.** The system maintainer must work immediately to code a fix to be installed before moving the software into production.

EXAMPLES:

- Potential impact of problem could be significant or minimal
- Problem affects CMS required reporting

Priority 4 Classification

Production:

The problem is an operator inconvenience or annoyance, which does not affect a required mission essential capability. The system maintainer must request approval to code and implement a fix from its CMS maintenance lead/(GTL).

EXAMPLES:

- Problems affects non-mission critical functions
- Operational procedure with workload impact that should be automated
- Impact of problem is minimal
- Correctable by contractor workaround*

Beta/User Acceptance Testing:

The problem is a test inconvenience or annoyance, which does not affect a required mission essential or test capability. If moved into the production environment before correcting, an acceptable workaround could be instituted to prevent the inconvenience.** The system maintainer should work immediately to code a fix to be installed before moving the software into production.

EXAMPLES:

- Problem affects non-mission critical functions
- Operational procedure with workload impact that should be automated
- Impact of problem is minimal
- Correctable by contractor workaround*

Priority 5 Classification

Production:

All other documented system problems. These could include operator errors, an inability to reproduce the reported problem, a problem with insufficient information, or documentation errors. The system maintainer should request approval from the CMS maintenance lead/(GTL) before coding and implementing any system enhancements.

EXAMPLES:

- A/B and DME MACs

  requested enhancements
- Documentation errors (i.e. Business requirements)
- Problem affects non-mission critical functions
- Minimal impact

Beta/User Acceptance Testing:

All other documented system test problems. These could include operator errors, an inability to reproduce the reported problem, a problem with insufficient information, or test documentation errors. The system maintainer should work to correct these issues as soon as possible but any system enhancements should be discussed with the CMS maintenance lead/(GTL).

Examples:

- Test region or processing enhancements
- Test documentation errors (i.e. business requirements)
- Problem affects non-mission critical test functions
- Minimal impact

* An acceptable workaround is a temporary alternative solution to a confirmed problem in the shared system that will ensure the contractor is able to accomplish a mission critical capability. What makes the workaround “acceptable” is it must be agreeable to both the maintainer and contractor and does not cause an excessive burden to the contractor. If the maintainer and A/B and DME MACs cannot come to an agreement on what is “acceptable” the decision will be made by CMS.

** CMS does not recommend using workarounds in the test region in order to "pass" test cases. The institution of a workaround should be used in order to implement a CMS mandate where the system maintainer may not have time to adequately code a fix before the software is delivered for production installation.

Routine File Maintenance/Updates

CMS does not require pre-approval or special documentation of routine file maintenance/updates or other routine activities necessary for effective operation of the Medicare system, Medicare processes and/or testing (e.g., MR/UR screen updates, provider and beneficiary file updates). All contractors and data centers should continue with their normal file maintenance routines.

Testing Prior to Installation of CMS Approved Follow-up Releases

CMS explains expectation for each Medicare organization's testing responsibility (i.e., shared system maintainer testing, contractor testing, CWF host testing, Beta testing).

Information Required for Requesting CMS Approval

The following must be submitted to the CMS maintenance lead or project officer when requesting that a problem be implemented in a follow-up release. If the system maintainer already has a process in place for communicating system problems to CMS, that process may be used as long as all information below, at a minimum, is captured.

MAINTAINER NAME:

Problem Description:

Brief non-technical business description of the fix.

How Found:

Explain how the problem was found. Also explain why you believe it was not found by release testing.

Problem Impact:

This information is needed to determine the scope of the problem in terms of payments, provider types, beneficiaries, number of potential claims impacted, it a work around is available, etc.

Problem Priority Classification:

Is this problem prioritized as an emergency, 1, 2, 3, 4, or 5.

Release Options:

Explain the options for scheduling and implementing the fix.

Technical Recommendation for Release timing:

Explain the recommended timing for installing the release.

CMS System Maintenance Leads/(GTL)

Maintainers must forward schedules and documentation of all changes as required in the memorandum to your CMS maintenance lead/(GTL) as indicated below. If your current process is to forward this information to your project officer, continue to do so. Your CMS maintenance leads/(GTL) will advise you of backup staff.

40.2.1 - Implementing Validated Workarounds for Shared System Claims Processing by All Medicare DME MACs

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

Medicare A/B, DME contractors shall implement workarounds within the shared systems for problems when formally defined as a Priority 3 or Priority 4 without obtaining written permission from a Project Officer or Regional Office.

Shared system problems that are formally defined as a Priority 3 or a Priority 4 have acceptable workarounds which provide temporarily alternative solutions. In order for a A/B/DME MACs to implement a workaround, the shared system maintainer must first validate the problem, confirm that the workaround exists, is systematically viable and does not cause adverse effects. The implementation of such workarounds will eliminate delay in adjudication of Medicare claims and the payment to providers. Utilizing a Priority 3 or Priority 4 workaround shall not diminish the integrity of the shared systems and shall not include such actions as deactivating standard edits. The shared system's priorities are formally defined at Section 40.2 of this chapter.

40.3 - Shared System Testing Requirements for Shared System Maintainers, Single Testing Contractor (STC)/Beta Testers, and Medicare A/B, Durable Medical Equipment (DME) Administrative Contractors

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

Medicare requires implementation of a limited number of shared systems that must be used by all MACs for the administrative responsibilities of Traditional Medicare (Part A, B and DME claims). This eliminates the need for each A/B, DME MAC to repeat development of the base system as part of the complete system development life cycle (SDLC).

CMS requires that the shared system quarterly release be subjected to the complete testing life cycle prior to production release. The goal is to ensure that all changes function as intended and that the implementation of changes does not degrade or otherwise unintentionally affect existing system capability and function prior to implementation. This requires that the shared system be subjected to all levels and types of testing including unit testing, integration testing, systems testing, functional testing, interface testing, performance testing, regression testing, and operational testing. Definitions are provided in subsection 40.3.9.

The Shared System Maintainer and the Medicare A/B, DME MACs each have specific roles in testing the shared system quarterly release. Additionally, CMS contracts with a CMS Single Testing Contractor (STC) to act as a Beta tester for the FISS, MCS, VMS, and CWF shared systems respectively.

The CMS Single Testing Contractor (STC) assumes primary responsibility for testing the Medicare Shared Systems. The STC will be fully responsible for meeting the requirements of the Beta tester as outlined in Chapter 7, Section 40.3, including all subsections. STC interface testing with HIGLAS will be initiated as a shadow test on Part A. The STC shall also initiate Railroad Retirement Board (RRB) testing. Three Medicare Contractor numbers have been established to accommodate STC testing. They are 00888 for MCS systems testing, 00388 for FISS systems testing, and 44410 for DMEMAC systems testing.

This section identifies the testing responsibilities for each organization to ensure that each shared system quarterly release satisfies all CMS requirements. All organizations shall have processes in place to meet these requirements. Testing activities will generally begin 3 to 4 months in advance of the release date, particularly for shared system maintainers..

40.3.1 – Shared System Testing Requirements for Shared System Maintainers, Single Testing Contractor (STC), and DME MACs

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

Review subsection 40.3.9, Definitions, for a description of key testing terminology.

1. Maintainers of a Shared System shall plan and execute all the essential levels of testing. At a minimum this includes Unit testing, Integration testing, System testing, and Regression testing. Shared System Maintainers are also responsible for performing Interface Testing.

2. Beta testers may initiate testing at the integration level, but are primarily dedicated to testing at the system level, including regression testing. Beta testers are also responsible for performing Interface Testing, which includes full data exchanges between the shared system maintainers, and other systems.

3. Shared System Maintainers and Beta testers shall maintain a test environment that enables system-testing activities to replicate the production environment, as closely as required to effectively test. CMS provided all Beta testers with a date simulation tool to facilitate executing test cases with future dates (e.g., service dates, admission dates) without turning off edits or altering effective dates in the test environment.

40.3.2 – Minimum Testing Standards for Shared System Maintainers and the Single Testing Contractor (STC)/Beta Testers

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

1. The Shared System Maintainers (SSMs), and the Single Testing Contractor (STC) shall fully test the quarterly release to ensure it is ready to be elevated to production. For the quarterly release to be considered fully tested, all the requirements contained within the release must be tested. Shared System Maintainers (SSMs) and the Single Testing Contractor (STC) must be able to demonstrate the degree to which each discrete requirement within a CR has been tested and by which test cases. It is therefore mandatory that the testers maintain traceability between test cases and the discrete requirements being implemented in the release. Additionally, for each CR or transmittal under test, the Shared System Maintainers and the Single Testing Contractor (STC) must ensure that each discrete requirement specified in the **Business Requirements** section of any CR/transmittal has been fully tested. The Shared System Maintainers and the Single Testing Contractor (STC) shall specifically:

   - Maintain a repository of test requirements against which all test cases must be traced.

   - Prepare and execute a set of test cases that demonstrate the requirements were correctly implemented for all change requests within the quarterly release.

   - Maintain traceability between each test case and the requirement that the case was designed to test.

2. The Shared System Maintainers and Single Testing Contractor (STC) shall distinguish each test requirement with a unique Requirement Identifier. The Requirement Identifier must be a number or qualifier preceded by the CMS CR number and SSM CR number, separated by dashes. The format of the Requirement Identifier is: [SSM CR No.]-[CMS CR No.]-[Requirement No.], where:

   - SSM CR No. – is a number that identifies a CMS mandate or user change request under test. Free form text can also be used to identify changes not associated with maintainer CR numbers, e.g., “Regression” to indicate regression testing. Avoid spaces and use underscore symbol “_” instead.

Dashes not allowed within this number; they are reserved for separation between the individual numbers in the Requirement Identifier.

- CMS CR No. – is a minimum 4-digit number that identifies the CMS CR associated with the maintainer CR number. If no CMS CR is associated with a maintainer CR, use "0000". Dashes are not allowed. Avoid spaces and use underscore symbol "_" instead. Dashes not allowed within this number; they are reserved for separation between the individual numbers in the Requirement Identifier.

- Requirement No. – is a number that uniquely identifies the requirement with the CR. For any requirement taken from the Business Requirements section of a CMS CR, use the actual number from the Requirement # column. Do not repeat the CMS CR number. Dashes not allowed within this number; they are reserved for separation between the individual numbers in the Requirement Identifier.

  Example: Maintainer CR 22522 corresponds to CMS CR 2634. Business Requirement 2.8 was taken directly from CMS CR 2634. The Requirement Identifier would be 22522-2634-2.8.

3. The Shared System Maintainer and the Single Testing Contractor (STC) shall complete Test Case specifications that include specific input situations and the expected results associated with a single test purpose. Each test case specification must include the following:

   - A unique Test Case Identifier (which includes a cross-reference to the requirement in which the case is designed to test);

   - The specific objective or purpose of the case;

   - Input specifications (i.e., a description of the input situation[s]);

   - Output specifications (i.e., a description of the expected results); and

   - Intercase dependencies - in instances where the test results of one test case may impact other test cases, the test case specification must identify the other test case(s) and describe the relationship(s).

   Refer to section 40.3.10, Test Case Specification Standard, for the specific format required to electronically maintain test cases.

4. All Test Cases must contain a unique Test Case Identifier. The CMS standard for the Test Case Identifier is the Requirement Identifier, followed by a number that uniquely qualifies the test case specification, separated by a dash.

   The format of the Test Case Identifier is: [Requirement Identifier]-[Test Case Number], where:

   - Requirement Identifier – is the actual identifier of the requirement being testing by the case.

   - Test Case Number – is a number that uniquely qualifies the test case. This is generally a sequential number. This is necessary since more than one test case is often needed to test a single requirement.

Dashes not allowed within this number; they are reserved for separation between the individual numbers in the Test Case Identifier.

Example: Two test cases were developed to test the implementation of Requirement 22522-2634-2.8 (see example above). The unique Test Case Identifier for the two test cases would be 22522-2634-2.8-01 and 22522-2634-2.8-02.

5. The Shared System Maintainers and the Single Testing Contractor (STC) shall document and execute **both** positive and negative test cases to ensure the requirements of the release are correctly implemented.

   - Positive test cases are required to ensure that the system is directly fulfilling the requirements as specified. One or more positive test cases are required for each requirement. As an example, if a program mandate effects a change for services beginning on July 1, a positive test case would include service dates in July or later and validate that the actual mandate was correctly implemented.

   - Negative test cases test cases are required to ensure that the system does not perform an incorrect action. As an example, if a program mandate effects a change for services beginning on July 1, a negative test case would ensure that implementing the mandate did not negatively impact claims with service dates prior to July 1. Unlike positive test cases, a negative test case may not be applicable to every requirement within a CR. Additionally, although due diligence might necessitate a negative test case, the need may be mitigated by an existing case in your regression test set.

6. The Shared System Maintainers and the Single Testing Contractor (STC) shall document all test cases and the actual results for each test case electronically. Each test case and the associated results must be stored in a test management repository (i.e., Application Lifecycle Management (ALM) tool) and must at a minimum contain the data elements outlined in the CMS Test Case specification standard. See subsection 40.3.10 for the Test Case specification standard.

7. The Shared System Maintainers and the Single Testing Contractor (STC) shall maintain a test log that provides a record of each test execution. Test Log requirements may be fulfilled by correctly using the ALM "run" feature (or an equivalent tool or approach) as outlined in the Quarterly Release Test Management User Guide.

8. The Shared System Maintainers and the Single Testing Contractor (STC) shall execute a full regression test set on their system for every quarterly release. Each testing entity shall perform regression testing within their designated testing window as outlined in subsection 40.3.7, Timeframe Requirements.

9. The Shared System Maintainers and the Single Testing Contractor (STC) shall perform interface testing.

   - The Shared System Maintainers and the Single Testing Contractor (STC) shall validate that all output files are correctly created by their system. The SSMs and STC shall validate that their system can accept and correctly process all input files.

   - The Shared System Maintainers and the Single Testing Contractor (STC) shall perform interface testing that includes full data exchanges (both ways) between the shared system and any principal claims processing adjudication or financial system (e.g., the CWF and HIGLAS respectively). The STC is required to perform data exchanges with HIGLAS after HIGLAS is implemented at the STC's data center.

- The Shared System Maintainers and the Single Testing Contractor (STC) shall complete an integrated system test coordinating the maintenance of test data baselines, such as beneficiary data, with the CWF Beta tester.

40.3.3 – Testing Standards Applicable to all Beta Testers

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

1. Each shared system maintainer and each beta tester shall complete integrated testing with the CWF Beta tester, using coordinated beneficiary data, in the execution of their test cases. All test cases involving CWF functionality (related to claims adjudication) must be executed in an integrated test with the CWF. This requires full data exchanges between testing entities including:

    - Satellite files being sent from the shared systems to the CWF Beta tester; and

    - Response files being sent from the CWF to the shared system Beta testers.

2. Each Beta tester shall:

    - Utilize the standard CMS Test Management tool and repository to documents all test cases and results.

    - Follow the procedures outlined in the Quarterly Release Test Management User Guide in order to complete the documentation of test runs and results.

3. Each STC/beta tester shall review all Shared System Maintainer release documentation for completeness, accuracy, and usability. Any questions, problems, or issues with the documentation shall be forwarded to both the Shared System Maintainer and CMS.

4. Each STC/beta tester shall conduct performance testing to reasonably assure that the system provides acceptable response times, throughput rates, and processing windows and can accommodate production workloads.

5. The CMS testing requirements outlined in section 40.3 may require the STC/beta tester to test a specific type of bill, specialty, or claim situation for which they do not possess the required level of expertise. In these instances, the STC/beta tester must partner with a Medicare A/B, DME MAC that possesses both the expertise and capabilities to test the specialty or claim type. As an example, should the STC/beta tester not have the operational capability or expertise to process Home Health claims, they are expected to partner with an RHHI to complete the required HHA testing. Ultimately, the STC/beta tester is responsible for ensuring all test cases are exercised. Any partnerships that are established to complete the testing requirements shall be arranged and managed by the STC/beta tester.

40.3.5 – Part A/Part B (A/B) Durable Medical Equipment (DME) Medicare Administrative Contractor (MAC) (User) and the Single Testing Contractor (STC) Testing Requirements

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

A/B, DME MACs are not mandated to prepare and execute test cases that cover Medicare business requirements implemented within the base system in standard system and CWF quarterly releases. Shared System Maintainers and the STC are fully responsible for testing the base functionality. The A/B, DME MAC (users) shall test their local/unique components and conduct a limited, end-to-end, operational test.

1. A/B, DME MACs shall fully test their local components and processing rules prior to production implementation of the quarterly release. This testing is applicable for all local components and processing rules modified since the previous quarterly release.

   A. A/B, DME MACs shall test any system components they maintain and implement to support claims processing in addition to the base system. This includes front-end and back-end components such as those for EDI entry and translation, EDI outbound processing, and printing (e.g., MSN generation).

   B. A/B, DME MACs shall test changes they make to user control files, facilities, and tables in order to implement new Medicare policy or business rules. Examples of these facilities include but are not limited to auto adjudication facilities (e.g., ECPS and MCS SCF) and the MCS SPITAB.

   C. A/B, DME MAC shall fully test any shared system functionality that was:

      - Developed by the shared system maintainer solely for them, or
      - Developed by the shared system maintainer under a special project in which they were the exclusive participant.

      An example would be a MAC working with CMS on a special demonstration project. In this example the MAC shall fully test the shared system functionality that was implemented for the demonstration project.

2. A/B, DME MACs shall complete a limited end-to-end operational test that incorporates the shared system release, integrated with their other claims processing components. These components include the front-end for claims receipt, translators, the CWF, the financials, and back-end EDI and report generation. The test must ensure that processing is contiguous from claims entry, to claims adjudication, and ultimately remittance and Medicare Summary Notice generation. Through contiguous processing, the interfaces between all key claims processing components must be exercised. The banking system interfaces such as National Clearing House (NCH) transfers need not be exercised. The test is limited in the number of test cases that are required, since shared system maintainers and the STC are testing the base functionality of the shared system.

   A. A/B, DME MACs shall ensure that the integrated systems software can complete cycles without system abends and produce the expected output. The A/B, DME Medicare Contractor shall ensure their operational test:

      - Exercises all claims entry points not fully incorporated in the base system such as paper and EMC front-end components;
      - Includes all allowable standard electronic formats and versions;
      - Includes a variety of claims types; and
      - Includes all components that support their claims workload and interfaces to the shared system.

B. The operational test shall include the most recent shared system release received prior to the initiation of the test. A/B, DME MACs shall initiate the test as required to ensure its completion and the reporting of any problem prior to production implementation.

3. CMS strongly encourages the shared system user community to promote:

- Standardizing their system nationally,
- Centralizing any table maintenance that implements national Medicare policy at the system maintainer level, and
- Minimizing local variations.

4. A/B, DME MACs may perform additional testing on the shared system or duplicate Beta testing tasks as time permits. At the discretion of their Regional Offices, the A/B, DME MACs may be required to separately document any testing they perform in addition to their mandated testing.

5. A/B, DME MACs shall test any business rule or event with future dates that they code or set-up in their Auto Adjudication Software (AAS), prior to implementing the rule or event into production. Examples of AAS include but are not limited to ECPS, SCF, the Shack, and the Mill.

a. A/B MACS shall submit their date simulation recommendations to the FISS FWG in advance and participate in discussions at the FWG calls, as required to reach consensus on a date simulation schedule.

b. A/B MACS shall submit their date simulation recommendations to the MCS FWG in advance and participate in discussions at the FWG calls, as required to reach consensus on a date simulation schedule.

c. DME MACS shall submit their date simulation recommendations to the DMOP TAG or the CFMTAG in advance and participate in discussions at the TAG calls, as required to reach consensus on a date simulation schedule.

d. The FISS FWG, MCS FWG, and DMOP/CFM TAG shall use the "system date request process" to provide the VDC(s) their latest "run-date simulation" schedule for their FISS, MCS, and VMS UAT environments.

e. The FWG or DMOP TAG shall maintain their "run-date simulation" schedule for a **minimum** of 14 calendars days in advance. Here is an example of a "run-date simulation" schedule:

Calendar Date	System Run Date	Calendar Date	System Run Date	Calendar Date	System Run Date

10/8/2015	10/5/2015	10/17/2015	10/14/2015	10/26/2015	10/23/2015
10/9/2015	10/6/2015	10/18/2015	10/15/2015	10/27/2015	10/24/2015
10/10/2015	10/7/2015	10/19/2015	10/16/2015	10/28/2015	10/25/2015
10/11/2015	10/8/2015	10/20/2015	10/17/2015	10/29/2015	10/26/2015
10/12/2015	10/9/2015	10/21/2015	10/18/2015	10/30/2015	10/27/2015
10/13/2015	10/10/2015	10/22/2015	10/19/2015	10/31/2015	10/28/2015
10/14/2015	10/11/2015	10/23/2015	10/20/2015	10/1/2015	10/29/2015
10/15/2015	10/12/2015	10/24/2015	10/21/2015	10/2/2015	10/30/2015
10/16/2015	10/13/2015	10/25/2015	10/22/2015	10/3/2015	10/1/2015

f. The FISS FWG, MCS FWG, and DMOP/CFM TAG shall provide a copy of their "run-date simulation" schedule to their CWF test host.

g. The FISS FWG and MCS FWG shall provide a copy of their "run-date simulation" schedule to the HIGLAS test site as required to accommodate their HIGLAS enabled contractors.

40.3.6 – Testing Requirements Applicable to all CWF Data Centers (Hosts)

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

Each CWF data center or CWF sector host shall:

- Forward all satellite software and documentation to the satellites (contractors) to which they serve as the primary production host;
- Install the CWF quarterly release software in a designated test region;
- Make the designated test region available their satellites (users) for testing;
- Coordinate test data, such as beneficiary data, with the user testers;
- Process all satellite files submitted by the users and return all corresponding reply files generated for users;
- Report release problems to the CWF Maintainer and CMS;
- Verify with CMS that each of its satellites submitted at least one test file during user testing; and
- The CWF Host shall support future date testing by the MACs.

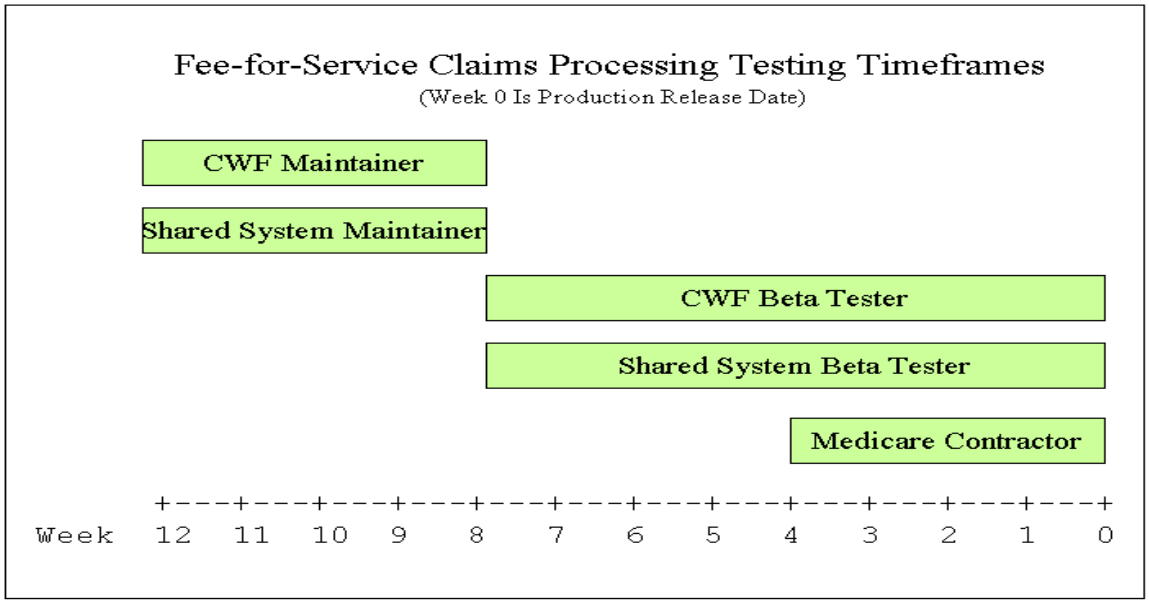
- The CWF Host shall review date simulation schedule submitted by the FISS FWG, MCS FWG, and DMOP TAG.

40.3.7 – Timeframe Requirements for all Testing Entities

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

The SSMs, STC, and MAC shall operate under the testing timeframes shown below for each quarterly release:

Timeframe Requirements for Testing Entities



1. The Medicare A/B, DME MAC or **User** testing period shall begin four weeks prior to production implementation.

2. The **Beta** testing period shall begin eight weeks prior to production implementation. The CWF and standard system Beta testers shall have an exclusive four-week testing timeframe prior to the initiation of user testing.

   - The Beta tester shall complete a functional System Test and Regression Test before the shared system is released to the User community.

   - Beta testing must also continue through the User testing period. The Beta tester may initiate performance testing during the user testing period.

3. Exclusive CWFM and SSM testing shall continue until Beta testing is initiated eight weeks prior to production implementation. The SSM and CWFM shall complete a Unit Test (on all components), Integration Test, System Test, and Regression Test prior to distributing the shared system release to the designated Beta tester. For all integration, system, and regression testing, the SSM shall use the most recent version of any third party or CMS provided software components (e.g., Pricer, OCE, MCE, Grouper) they are provided. The SSM shall continue testing beyond the exclusive maintainer-testing window due to the late receipt of some third party or CMS provided software components such as the Pricers and OCE.

40.3.8 – Testing Documentation Requirements

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

1. The SSM, STC, and MAC shall maintain documentation that fully demonstrates the requirements of this transmittal were met for each quarterly release. At a minimum the SSM, Beta tester, and MAC shall maintain the following test documentation to demonstrate full compliance:

    - Test Requirement and Test Case repository with traceability;
    - Test Log;
    - Test Status for the execution (run) of each test case (i.e., Pass, Fail, Not Run);
    - Actual Results for the run of each test case in which the actual results did not match the expected results; and
    - Documented proof of each test run, i.e., screen shots, scheduler job logs, etc.

2. The SSM, STC, and MAC shall:

    - Maintain all test documentation for the four quarterly releases prior to the current release under test. The documentation must be available for review by CMS (or its agent).
    - Document all software defects (problems) within the CMS specified repository such as INFOMAN.

3. The SSM shall communicate all confirmed software defects (problems) and fixes directly to CMS in writing through their CMS Maintenance Lead/(GTL) or other designee as specified by the CMS Project Officer.

4. A/B, DME MACs shall provide any testing documentation to their CMS regional office upon request.

Additional requirements for selected shared system maintainers, STC, and CWF hosts may be contained in these organizations' individual contracts. Electronic screen shots may be incorporated/attached into the results of ALM has proof of online results. Test Log requirements may be fulfilled by correctly using the ALM "run" feature as outlined in the Quarterly Release Test Management User Guide.

40.3.9 – Definitions

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

These definitions are provided to ensure common understanding.

Base Shared System - The FISS, MCS, VMS, or CWF system, which is routinely released by the Shared System Maintainers to their respective user community prior to any user customization. This includes all components released by the Shared System Maintainer, including but not limited to the claim adjudication subsystem, the financial subsystems, and other integrated components (i.e., Pricer, OCE, MCE, Grouper).

Functional Testing – Testing to ensure that the functional requirements have been met.

Integration Testing – Testing combinations of interacting software components that make up parts of a system.

Interface Testing – Testing conducted to evaluate whether subsystems or systems pass data and control to one another correctly.

Local Components – A Local Component as referenced in section 40.3 is any component or module that supports Medicare claims processing, but is not part of the Base System and is under the control and maintenance of the MAC.

Maintainer – The Maintainer is an entity to which CMS directly contracts to maintain a Medicare claim processing shared system (FISS, MCS, VMS, or the Common Working File (CWF) system). The Maintainer, as referenced in section 40.3, does not refer to an entity to which a A/B or DME MAC subcontracts to operate their data center or perform other claim processing support activities.

Operational Testing – Testing conducted to evaluate a system in its operational environment. Testing to ensure that the aggregate operational systems and their interfaces can be operated securely with the instructions provided.

Performance Testing – Testing that applies heavy transaction and processing loads to the system to ensure that response times, throughput rates, and processing windows remain acceptable and can accommodate production workloads.

Regression Testing – Testing conducted on a system or components to verify that modifications have not caused unintended effects and that the system or components still complies with its requirements.

Regression Test Set – A set of selectable test cases designed to exercise a system over its functional capabilities and assure that it still works properly after changes have been applied.

Requirement Identifier – A unique number assigned to each requirement comprised of the Standard System Maintainer CR Number, the CMS CR Number, and an alphanumeric element to uniquely qualify each requirement. For testing purposes CMS requires that each Test Case Identifier incorporate the Requirement Identifier to which it is traced.

Stress Testing – Testing that applies a steadily increasing load to the system until it reaches the point where performance degrades to unacceptable levels.

System Testing – Testing to discover any incorrect implementation of the requirements or incompatibilities in the software/hardware environment. System testing includes functional testing, performance testing, and operational testing.

Test Case Specification – A description of an input situation and of the required results associated with a specific test objective or purpose.

Test Case Identifier – A unique identifier assigned to each test case.

Test Log – A chronological record of relevant detail about the execution of tests. Relevant details include run date, run time, test status, and actual results.

Test Requirement - A specific requirement that is under test and to which one or more test cases are traced. Test requirements may be derived from various types of requirements i.e., business functional requirements, performance requirements etc. Note: Any well-written requirement that is “testable” may be considered a Test Requirement. Any requirement contained in the Business Requirements section of a CR or transmittal, also constitutes a test requirement.

Test Set – A collection of test cases that have a common usage.

Unit Testing – The testing of individual units (i.e., software components, modules) or groups of related units. It is the lowest level of testing and is usually performed by programmers. Unit testing may be both functional (requirements oriented) and structural (i.e. logic oriented, code coverage oriented).

40.3.10 - Test Case Specification Standard

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

Purpose: This standard establishes a controlled outline for the contents and presentation of a Test Case Specification used by the shared system maintainers and the STC .

Applicability: This standard is applicable to all Test Case Specifications developed by the shared system maintainers and the STC.

Data Element	Description	Allowable Values or Format	Comments
Test Case Specification Identifier	Multi-part indicator that uniquely identifies the test case specification.	See Test Case Specification Identifier Standard.	
Test Purpose	A free form field that captures the intent of the test and identifies any key components of the test, e.g., specific codes.	See attached example.	
Input Specification	A free form field that captures critical information used to exercise the system functionality. Information could be grouped into the following topics: Claim Data Requirements Claims History Beneficiary Information Provider Information	See attached example.	
Intercase Dependencies (Predecessor Transaction Identifier)	The test case specification identifier of the transaction that must be entered into and processed by the system prior to processing the transaction described by the test case specification.	See Test Case Specification Identifier Standard.	
Output Specification	A free form declarative statement that identifies the expected results from performing all the steps, as a collection, within the test.		

Test Type	A one-character indicator to identify whether the test is positive or negative.	P = Positive Test N = Negative Test	**ALM** **Plan Tab** **(Required User Defined Fields)**
Originator	A one-character indicator to identify the originating entity (designer) of the test case.	B = Beta C = CMS/QRTM M = Maintainer	
Test Status	Summary indicator for a test case.	PS = Passed FA= Failed NR = Not Run IN = Incomplete ID = Invalid Data IC = Invalid Case	**Required** **Test Execution (Run)** **Elements**
Test Results	Free form declarative statement of actual results for a test case when the actual results do not match the expected results.		

Optional Information: Industry best practices demonstrate that additional granularity may be necessary to document discrete key test actions that should be executed and documented. These items are referred to as test steps. A test case specification may have one or more test steps. When documenting test steps, the following standard applies:

Step Number	Unique identifier for each test step.	"Step n" Where "n" is a sequential counter for each step starting at 1. There is at least one test step in each test case specification, but usually contains multiple test steps.	**Optional** **Test Case Elements**
Step Description	A free form declarative statement that identifies the action taken to perform the test. The step description statement usually begins with a verb.		

Expected Step Results	A free form declarative statement that identifies the expected results from performing the associated step description.		

Example #1

	Test Case Identifier	4419-2825-5.2-001
	Test Purpose	To confirm that the A/B MAC (A) claims processing systems accept, process, and assign reason code 30 (Payment adjusted because the patient has not met the required eligibility, spend down, waiting or residency requirements) to Inpatient Hospital claims submitted on Type of bill (TOB) 111 (Hospital Inpatient Part A; admit through discharge) with Dates of Service (DOS) on 01/01/2004 when a beneficiary is not lawfully present in the United States.
Input Specification	**Claims History**	None
	Beneficiary Information	Beneficiary must be unlawfully present in United States. Beneficiary elected English as primary language
	Provider Information	Provider Number Range = XX0001-XX0999
	Claim Data Requirements	TOB = 111 DOS = 01/01/2004
	Intercase Dependencies	None
	Output Specification	Claim will be assigned reason code 30 indicating beneficiary is not lawfully present in the United States, generating MSN message 5.7 (Medicare payment may not be made for the item or service because on the date of service, you were not lawfully present in the United States).
	Test Type	P
	Originator	C
	Test Status	PS
	Test Results	Claim was assigned appropriate reason code

40.3.11 - *Next Generation Desktop (NGD)* Requirements

(Rev.97, Issued: 01-15-16, Effective: 09-21-15, Implementation: 09-21-15)

CMS has fully deployed the Next Generation Desktop (NGD) to the contractors' beneficiary customer service contact centers. The NGD is a multiple call center application that is used by Medicare Customer Service Representatives (CSRs) to answer inquiries and perform operations on behalf of CMS beneficiaries and the American public.

The NGD is designed to pull customer service-needed information into a common desktop application. As such, the NGD requires data exchange with CMS shared systems (VMS, CWF, FISS, MCS) and standard systems (Enrollment Database (EDB)/Master Beneficiary Database (MBD), Master Beneficiary Record (MBR), Group Health Plan (GHP)/Maricopa Managed Care System (MMCS). Note: NGD may integrate with additional systems as future releases are developed.

Because NGD integrates with the shared systems, periodic changes will be made to the NGD Integration Layer as a result of the shared systems quarterly release process. The NGD maintainer will be required to update NGD in shares systems quarterly releases (Jan, Apr, July, Oct.) The NGD maintainer will be required to perform the various activities associated with changes to the NGD (i.e., unit and system testing). In addition to the shared systems quarterly release schedule, the NGD will adhere to a separate quarterly functional release process for NGD-specific updates and defect correction.

The NGD maintainer shall follow all of the requirements identified in Section 40.3 for the shared system maintainers except as indicated below:

1. Section 40.3.1 Maintainers and Beta Testers –Required Levels of Testing, #3 is not applicable to NGD Beta testers.

2. Section 40.3.2 (#2) Minimum Testing Standards for Maintainers and Beta Testers, for NGD naming conventions, the NGD Maintainer should refer to the NGD test Plan.

3. Section 40.3.2 (#4) Minimum Testing Standards for Maintainers and Beta Testers, for NGD test case identifiers, the NGD maintainer should refer to the NGD System Test Plan.

4. Section 40.3.7 Timeframe Requirements for Testing Entities – NGD testing timeframes are as follows:

- The official NGD User Acceptance testing period conducted by the VCS Support Contractor shall begin 2 weeks prior to production implementation.

- The overall NGD testing period shall begin 3 weeks prior to production implementation. The NGD VCS Support testers shall have an exclusive 1 week testing timeframe prior to the initiation of user testing (Validation testing)

-The User Acceptance tester shall complete a functional System Test and Regression Test before the system is released to the User community.

- Exclusive NGD System Maintainer testing shall continue until User Acceptance testing is initiated 3 weeks prior to production implementation. The NGD Maintainer shall complete a Unit Test (on all components), Integration Test, System Test, and Regression Test prior to distributing the shared system release to the designated Beta Tester.

5. Section 40.3.8 Testing Documentation Requirements (#2) For NGD, documentation of all software defects (problems) should be through ClearQuest.

5. Section 40.3.8 Testing Documentation Requirements (#2) For NGD, documentation of all software defects (problems) should be through ClearQuest.