Existing Product Intake Program (EPIP)

Patch MC\*2.3\*46

Remediation Plan



Department of Veterans Affairs

**February 2017**

Version 3.0

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 02/15/2017 | 3.0 | Updated routine information in section 5.3, Appendix A, and Appendix B | EPIP Project Team |
| 12/22/2016 | 2.0 | Updated entire document | EPIP Project Team |
| 11/28/2016 | 1.0 | Initial version | EPIP Project Team |

Table of Contents

[1. Introduction 1](#_Toc473617140)

[2. Purpose 1](#_Toc473617141)

[3. Patch Description 1](#_Toc473617142)

[3.1. Needs and Requirements 1](#_Toc473617143)

[4. Points of Contact 2](#_Toc473617144)

[5. Code Remediation 2](#_Toc473617145)

[5.1. Standards and Conventions 2](#_Toc473617146)

[5.2. Review and Analysis 2](#_Toc473617147)

[5.3. Coding Changes 2](#_Toc473617148)

[6. Testing 2](#_Toc473617149)

[6.1. Test Plan 3](#_Toc473617150)

[6.2. Test Environment 3](#_Toc473617151)

[6.3. Test Readiness Review 3](#_Toc473617152)

[6.4. Testing Phases 3](#_Toc473617153)

[6.4.1. Unit Testing 3](#_Toc473617154)

[6.4.2. Component Integration and Systems Testing (CI/ST) 4](#_Toc473617155)

[6.4.3. Functional Testing 4](#_Toc473617156)

[6.4.4. Regression Testing 4](#_Toc473617157)

[6.4.5. Section 508 Compliance Testing 4](#_Toc473617158)

[7. Documentation Remediation 4](#_Toc473617159)

[7.1. User Guides 4](#_Toc473617160)

[7.2. Installation Guides 5](#_Toc473617161)

[7.3. Technical Manuals 5](#_Toc473617162)

[7.4. Operations Manuals 5](#_Toc473617163)

[8. Project Reporting 5](#_Toc473617164)

[9. Project Schedule 5](#_Toc473617165)

[10. Deployment 5](#_Toc473617166)

[11. Sustainment Requirements 5](#_Toc473617167)

[12. Maintenance and Knowledge Transfer 5](#_Toc473617168)

[Appendix A: XINDEX Listing for MUMPS Code Changes 6](#_Toc473617169)

[Appendix B: Source Code Changes 9](#_Toc473617170)

# Introduction

The Department of Veterans Affairs (VA) currently utilizes the Veterans Health Information Systems and Technology Architecture (VistA) suite of applications to provide clinical, financial, infrastructure, and management tools. The process of advancing “Class 3” field-developed VistA software to “Class 1” nationally-distributed status is referred to as the Existing Product Intake Program (EPIP). The VA’s goal is to supplement ongoing activities associated with evaluating and advancing field-developed software to a state that meets national standards and facilitates release for Veterans Health Administration (VHA)-wide use.

# Purpose

The purpose of this document is to fully describe the remediation plan to be used for the successful remediation of the intake product code to be deployed as patch MC\*2.3\*46. This patch addresses the following NSR:

* NSR 20160418 *Fix PFT Report Calculations*

This NSR has been implemented locally at the VA Medical Centers in Milwaukee, WI and Washington, DC.

This document addresses the schedule, code remediation, testing, documentation, and delivery of this remediation effort.

# Patch Description

Patch MC\*2.3\*46 amends code from VistA patch MC\*2.3\*25 that provided invalid diffusion calculations (inflated calculation of percent of predicted) in the Pulmonary Function Test Report. These erroneous calculations made significantly abnormal Diffusing Capacity of the Lungs for Carbon Monoxide (DLCO) measurements appear as if they were normal, endangering the health of patients.

Following release of MC\*2.3\*25, the Office of Information & Technology (OI&T) developed code to correct this calculation error and provided it to some VA sites. The code was supposed to be released nationally in a Medicine package patch; however, the patch was not released because the Medicine package was placed in Maintenance.

## Needs and Requirements

The Needs and Requirements for the NSR addressed in this remediation are:

NSR 20160418 *Fix PFT Report Calculations*:

* NEED 720577: Pulmonary Calculations - As a provider I need automatic pulmonary calculations from the Pulmonary Function Tests performed on my patients so that I can have accurate data to use when providing care to my patients.
* REQUIREMENT 720610: Specific Values For the PFT - As a clinician, I need the values for the Corr DLCO for HB & COHB on the PFT Report to accurately report abnormal values in order to ensure patient safety.

# Points of Contact

The VA Point of Contact (POC) for NSR 20160418 *Fix PFT Report Calculations* is

# Code Remediation

Leidos will review and analyze the intake product code for compliance with coding standards, pointers, shared tables, dependencies, and any interference with VistA systems.

## Standards and Conventions

Leidos will reference the website for applicable documents and will adhere to VA standards to complete the analysis of this intake product. The output of the VA XINDEX utility will be used to analyze the MUMPS source code and document the affected routines (see Appendix A).

The MUMPS coding standards website <http://71.174.62.16/Demo/AnnoStd> will also be used to ensure that the remediated code conforms to VA standards.

## Review and Analysis

Review and analysis of this intake product involves two parts: 1) verification that the source code changes specified in this document provide the desired effect, and 2) verification that the source code changes do not adversely affect any other VistA or CPRS functionality.

Testing will be performed to validate that the intended effect of these products is implemented, and that no other VistA or CPRS Graphical User Interface (GUI) functionality is adversely affected.

## Coding Changes

The coding changes required for NSR 20160418 *Fix PFT Report Calculations* are in the following MUMPS routines:

**Modified routines:** MCPFTP2A

**New routines:** None

A detailed analysis of the coding changes is provided in Appendix B.

# Testing

Leidos will perform all testing-related activities to ensure that the remediated code meets the expectations of the VA business owner.

## Test Plan

Leidos will configure the test environment, provide code modifications and end-to-end testing, and deliver applicable testing documentation, following Veteran-focused Integration Process (VIP) guidelines..

The Leidos developer will modify the software pursuant to the VA standards defined in the *Standards and Conventions* section of this document, and will conduct full unit testing of the functionality and verify performance of all software code before it is released to Leidos SQA. SQA will then perform all applicable testing types as described in the *Testing Phases* section of this document. The developer and SQA will resolve problems and address issues as they arise during testing and will document issues using the Rational Team Concert (RTC) defect tracking tool.

## Test Environment

Within five working days of approval of this Remediation Plan, the developer will configure the development/test environment on an Austin Information Technology Center (AITC) server or other VA-approved development/test environment used for this intake product and install the remediated Kernel Installation and Distribution System (KIDS) build. The environment will be restored to its original baseline state by the VistA system administrator after development testing is completed, followed by installation of the remediated software.

Upon notification from the developer of test environment readiness, SQA will commence with planned testing activities. The SQA test execution and reporting documentation will reside in the Rational Quality Manager (RQM) “EPIP” Project. In order to perform testing of this VistA modification, the following tools will be leveraged: RQM, Reflections emulator, CPRS GUI v31 (1.0.30.75), and SnagIt.

## Test Readiness Review

Leidos will conduct a Test Readiness Review (TRR) at the conclusion of unit testing to verify the contents of the software to be tested, the test schedule, test environments, test participants, and associated logistics. Leidos will provide an agenda prior to the TRR and written minutes after completion of the TRR, in accordance with the Performance Work Statement (PWS).

## Testing Phases

Leidos will perform development and SQA testing activities in phases, and will provide all required testing documentation.

### Unit Testing

The developer will conduct unit testing of individual units of source code to determine if they are fit for use.

### Component Integration and Systems Testing (CI/ST)

Component integration and systems testing will be conducted by SQA to ensure that connectivity to the VistA application exists and is functioning normally. SQA will record Passed/Failed outcomes and capture displayed content to document the system testing effort.

### Functional Testing

Functional testing will be performed by SQA to test the code modifications. This testing will ensure that the software functionality is in alignment with the Government Furnished Information. SQA will record Passed/Failed outcomes and capture displayed content to document the functional testing effort.

### Regression Testing

Regression Testing will be performed by SQA to ensure that the remediated code does not introduce errors to existing functionality. The regression test framework will be kept up-to-date with manual test cases and test scripts defining the inputs and expected outcomes. SQA will record Passed/Failed outcomes and capture displayed content to document the regression testing effort.

### Section 508 Compliance Testing

Section 508 Testing will be performed on VistA and CPRS code when new CPRS GUI changes are introduced by the developer. The VA-recommended Assistive Technology tool, JAWS, will be used to conduct the 508 testing. Test results and related documentation will be submitted to the VA Section 508 team in accordance with the VA 508 testing requirements. Defects found during testing will be assessed and remediated by the developer.

# Documentation Remediation

Leidos will review existing VA documentation for possible impact as a result of this remediation effort, and will make updates where applicable.

To determine the existing VA documentation that requires modification, Leidos will conduct a thorough review of the documents currently available from the VA Software Document Library (VDL) located at Keyword searches using terms relevant to this remediation effort will be used to identify documents that might be impacted; those documents were will then be reviewed in their entirety for any needed revisions.

The following sections outline the VDL documents to be revised for this remediation.

## User Guides

No User Guides require revision as a result of this modification.

## Installation Guides

The *National Patch Module Patch Description* document for this remediation will provide the procedure for installing KIDS packages migrated from the test environment to the VA   
Pre-Production environments. Therefore, no Installation Guides will be updated.

## Technical Manuals

No Technical Manuals require revision as a result of this modification.

## Operations Manuals

No Operations Manuals require revision as a result of this modification.

# Project Reporting

Leidos will provide interim progress updates during daily Scrum calls and weekly management calls with VA representatives.

# Project Schedule

Leidos will follow the Scrum Agile methodology for software development. It is anticipated that this patch will require five 2-week sprints.

# Deployment

Leidos will create a KIDS package containing the software changes necessary to fulfill the requirements for this remediation effort. A KIDS package, along with all related documentation, will be delivered to the Contracting Office Representative (COR) for acceptance. If accepted, the KIDS package can then be released for national VA consumption; otherwise, Leidos will correct any defects found and repeat the necessary remediation activities.

# Sustainment Requirements

Leidos will provide maintenance support for 60 days to the VA to support the final Class 1 product after it is nationally released.

# Maintenance and Knowledge Transfer

To facilitate continuous process improvement, Leidos will deliver *Sprint Review and Retrospective* slides and a *Lessons Learned* *Report* to VA upon completion of the final sprint.

XINDEX Listing for MUMPS Code Changes

The XINDEX tool is the standard tool used by the VA to analyze MUMPS source code. Following is a listing of the results of the XINDEX analysis of the affected routines.

V. A. C R O S S R E F E R E N C E R 7.3

[2008 VA Standards & Conventions]

UCI: VISTA CPU: ROU Jan 17, 2017@16:10:25

The BUILD file Data Dictionaries are being processed.

The option and function files are being processed.

Routines are being processed.

Routines: 1 Faux Routines: 0

NURSCPLC

--- CROSS REFERENCING ---

Press return to continue:

Compiled list of Errors and Warnings Jan 17, 2017@16:10:25 page 1

No errors or warnings to report

--- Routine Detail ---

\*\*\*\*\* INDEX OF NURSCPLC \*\*\*\*\*

Local Variables Line Occurrences ( >> not killed explicitly)

( \* Changed ! Killed ~ Newed)

>> DFN START+2\*,START+3\*,SORT+10,NQ\*,PRINT1+1,PRINT1+2

>> DIC(0 EN2+4\*

DT HEADER+6

IO START+1

IOF START+4,HEADER+4

IOSL PRINT1+2

IOST START+2,START+3,START+4,START+5,HEADER+1,HEADER+4

>> N1 SORT+6\*,SORT+10,NP\*,NQ,PRINT1+2

>> NBED SORT+6\*,SORT+10,NO\*,NP,NQ,PRINT1+2

>> NCOPY START+4

>> NL1 HEADER+2,PRINT+1\*,PRINT+2\*,PRINT+5\*,NN\*

>> NPWARD START+2\*

>> NSEC PRINT1+1\*,PRINT1+2

>> NURCAT PRINT1+2,FNDCLAS\*

>> NURFAC SORT+3,PRINT+1,PRINT+6

>> NURFAC(1 SORT+3,PRINT+1,PRINT+6

>> NURFAC(2 SORT+1\*,SORT+3,SORT+8,SORT+9,HEADER+5,PRINT+1\*,PRINT+6\*

PRINT+9\*,NM,NN

>> NURHOSP START+2,HEADER+5

>> NURI START+4\*

>> NURIEN START+2\*,START+3\*,SORT+1,SORT+2

>> NURMDSW EN2+2\*,EN2+4,EN2+5,SORT+3,HEADER+5

>> NUROUT EN2+4,EN2+5,EN2+6,EN2+7\*,START+1\*,HEADER+1,PRINT+9,NM,NN,NO,NP

NQ,PRINT1+2

>> NURPAGE START+1\*,START+4\*,HEADER+4\*,HEADER+6

>> NURPLCSR EN2+5\*

NURPLSCR EN2+4\*!,EN2+5!

>> NURPLSW EN2+2\*,EN2+5,SORT+4,PROD

>> NURPROG SORT+4,PRINT+1,PRINT+2,PRINT+6

>> NURPROG(1 SORT+4,PRINT+1,PRINT+2,PRINT+6

>> NURPROG(4 SORT+2\*,SORT+4,SORT+8,SORT+9,PROD,PRINT+1\*,PRINT+2\*,PRINT+6\*

NM\*,NN

>> NURQUEUE EN2+2\*,HEADER+1

>> NURQUIT EN2+2\*,EN2+7

>> NURSCLAS FNDCLAS

>> NURSCLAS("CL"

FNDCLAS\*

>> NURSHD HEADER+2\*,HEADER+6

>> NURSNLOC PRINT+1,PRINT+2,PRINT+3

>> NURSNLOC( START+3,PRINT+2,PRINT+5

>> NURSORT SORT+7\*,SORT+9\*,NN\*,NO,NP,NQ,PRINT1+2

>> NURSW1 START+1\*,START+4\*,HEADER+1,HEADER+3\*,PRINT+2\*,PRINT+3\*,PRINT+6\*

PRINT+9\*,PRINT1+2

>> NURSWARD START+2\*,START+3\*,SORT+8,SORT+9

>> NURSX PRINT1+2

>> NURSZAP EN2+2\*

>> NURX PRINT+4\*

>> NURY PRINT+4\*

>> NURZ PRINT+4\*

>> POP EN2+8

>> SSN PRINT1\*,PRINT1+2

U SORT+2

>> VA("BID" PRINT1

VADM SORT+11!

VADM(1 SORT+6

VAIN SORT+11!

VAIN(5 SORT+6

X SORT+8~\*,SORT+9\*,SORT+10

>> ZTDESC EN2+8\*

>> ZTRTN EN2+8\*

>> ZTSK EN2+8

Global Variables ( \* Changed ! Killed)

^DIC(213.9 EN2+1

^DPT( SORT+10

^NURSA(214.6 FNDCLAS

^NURSF(211.4 SORT+2

^NURSF(213.3 PRINT1+1

^NURSF(214 START+2,START+3,PRINT1+1

^TMP($J START+1!,SORT+8,SORT+9\*,SORT+10\*,HEADER+2,PRINT+1

PRINT+2,PRINT+3,PRINT+4,PRINT+5,PRINT+9,NM,NN,NO,NP,NQ

PRINT1+2

Naked Globals

^(0 PRINT1+1

Cache Objects

NONE

Marked Items

NONE

Label References

FNDCLAS PRINT1+1

HEADER PRINT+1,PRINT+2,PRINT+6,NN,PRINT1+2

NM PRINT+9

NN NM

NO NN

NP NO

NQ NP

PRINT START+4,START+5

PRINT1 NQ

QUIT EN2+4,EN2+5,EN2+6,EN2+7,EN2+8

SORT START+2,START+3

External References

$$GET1^DIQ SORT+2

EN1^NURSAGSP EN2+6

EN5^NURSAGSP EN2+4

EN9^NURSAGSP EN2+3

PRD^NURSAGSP EN2+5

^NURSAPCH PRINT1

EN6^NURSAUTL START+2

EN2^NURSCUTL FNDCLAS

EN6^NURSCUTL FNDCLAS

^NURSKILL QUIT+1

EN6^NURSUT0 EN2+7

EN7^NURSUT0 EN2+8

CLOSE^NURSUT1 QUIT+1

ENDPG^NURSUT1 HEADER+1,PRINT+3

NODATA^NURSUT1 PRINT+2,PRINT+6

$$CNTR^NURSUT2 HEADER+5,PROD

$$EN12^NURSUT3 SORT+1

1^VADPT SORT+5

DEM^VADPT PRINT1

$$REPEAT^XLFSTR HEADER+8,PROD

\*\*\*\*\* END \*\*\*\*\*

Source Code Changes

This appendix displays the VistA code before and after the updates required for this code modification were implemented. The following routines were affected:

**Modified routines:** MCPFTP2A

**New routines:** None

**MCPFTP2A**

**Before:**

HB G ABG:$P(MCHB,U)="" S MCHB=$G(^MCAR(700.2,MCHB,0)) G ABG:MCHB=""

 S HB="",I=0 F  S I=$O(^MCAR(700,MCARGDA,6,I)) Q:I'?1N.N  I $D(^(I,0)),$P(^(0),U,2) S HB=$P(^(0),U,2)

 ;G ABG:HB'<11,ABG:'HB X "S MCHB="\_$P(MCHB,U)

 G ABG:'HB X "S MCHB="\_$P(MCHB,U)

 ;W !,"Corrected DLCO for HB: ",$J(MCHB,8,2)

 W !,"Corr DLCO for HB & COHB:",?19,$J(MCIDL,6,2),?32,$J(MCHB,8,2),?42,$S(MCIDL'=0:$J(MCHB/MCIDL\*100,8,1),1:"")

ABG K HB,COHB,MCHB,MCCOHB G ^MCPFTP3

PRTLINE S MCP1=$G(MCP1),MCP2=$G(MCP2)

 W !,?5,MEAS,?15,UNITS,?25,$S(PRED:$J(PRED,MCLNG,MCDL),1:""),?35,$J(ACT,MCLNG,MCDL),?45,$S(PRED:$J(ACT/PRED\*100,5,1),1:"") W:$P(MCP1,U,PC) ?55,$J($P(MCP1,U,PC),MCLNG,MCDL) W:$P(MCP2,U,PC) ?65,$J($P(MCP2,U,PC),MCLNG,MCDL)

 W:(CI95)&(CI95'=PRED) ?72,$J(CI95,6,2) X MCFF Q

MCFF1 X MCFF Q

**After:**

HB G ABG:$P(MCHB,U)="" S MCHB=$G(^MCAR(700.2,MCHB,0)) G ABG:MCHB=""

 S HB="",I=0 F  S I=$O(^MCAR(700,MCARGDA,6,I)) Q:I'?1N.N  I $D(^(I,0)),$P(^(0),U,2) S HB=$P(^(0),U,2)

 G ABG:'HB X "S MCHB="\_$P(MCHB,U)

 W !,"Corr DLCO for HB & COHB:",?19,$J(PRED,6,2),?32,$J(MCHB,8,2),?42,$S(+PRED'=0:$J(MCHB/PRED\*100,8,1),1:"")

ABG K HB,COHB,MCHB,MCCOHB G ^MCPFTP3

PRTLINE S MCP1=$G(MCP1),MCP2=$G(MCP2)

 W !,?5,MEAS,?15,UNITS,?25,$S(PRED:$J(PRED,MCLNG,MCDL),1:""),?35,$J(ACT,MCLNG,MCDL),?45,$S(PRED:$J(ACT/PRED\*100,5,1),1:"") W:$P(MCP1,U,PC) ?55,$J($P(MCP1,U,PC),MCLNG,MCDL) W:$P(MCP2,U,PC) ?65,$J($P(MCP2,U,PC),MCLNG,MCDL)

 W:(CI95)&(CI95'=PRED) ?72,$J(CI95,6,2) X MCFF Q

MCFF1 X MCFF Q