DATS

Development Applications and Technical Services

Centers for Medicare & Medicaid Services

Electronic Submission of Medical Documentation (esMD)

Review Contractor (RC) Client Java User Guide and Installation Handbook

Version 1.1 Final

05/09/2016

Document Number: DATS_RCClient_Java_Imp_Gd
Contract Number: HHSM-500-2007-00024I

Table of Contents

1. Introduction 1

1.1 Overview of the esMD 1

1.1.1 The esMD Claim Review Contractors 1

1.2 System Overview 2

1.3 System Requirements 3

1.4 RC Client Overview 3

1.4.1 RC Client Pull/Push Functionality 4

1.4.2 RC Client Application Overview 4

1.4.3 RC Client Operation Overview 5

2. Overview of How This Document is Structured 8

3. How to Start the RC Client and Log In 10

4. How to Enter a Decision on the Review Decision Response to Prior Authorization (PA) Request Tab 12

4.1 How to Enter an A-Affirmed Decision 12

4.2 How to Enter a M-Affirmed with Change Decision 17

4.3 How to Enter a N-Non Affirmed Decision 28

5. How to Enter an Error Code on the Error Response to PA Request Tab 34

6. How to Submit an Inbound Submission Error on the Administrative Error Response to Inbound Submissions Tab 39

7. How to Verify Connection to TIBCO MFT server, Using the Advanced/Debugging Tab 42

8. System Requirements 44

8.1 Processor 44

8.2 Disk Space 44

8.3 Memory 44

8.4 Permissions 44

8.5 Network 44

8.6 Java Framework 44

8.7 Libraries 44

9. How to Install and Configure a Java Version of RC Client 45

9.1 Out-of-the-Box 45

9.1.1 Keystore Set Up 45

9.1.2 Integrity Verification 46

9.1.3 Java Cryptography Extension (JCE) Policy Update 46

9.1.4 Configuring the RC Client ... 48
9.1.5 Running the RC Client ... 51
9.2 Custom RC Client ... 51

10. TIBCO® MFT File Transfers ... 53

11. XML Messages ... 55

11.1 Inbound ... 55
11.1.1 Payload Files ... 55
11.1.2 Flat File Rendering (FFR) ... 55
11.1.3 Cover Sheet ... 56
11.1.4 Metadata File ... 59
11.1.5 Pickup HIH Status Response ... 61
11.1.6 Pickup Validation Error Response ... 61
11.1.7 Administrative Error HIH Status Response ... 62
11.1.8 Administrative Error Response Validation Error ... 62
11.1.9 Virus Scan Error Response ... 63
11.1.10 PA Review Result HIH Status Response ... 63
11.1.11 PA Review Result Validation Error Response ... 64
11.2 Outbound ... 65
11.2.1 Pickup Notification ... 65
11.2.2 Error Pickup Notification ... 65
11.2.3 Review Decision Response to PA Request ... 66
11.2.4 Error Response to PA request ... 69
11.2.5 Administrative Error Response to Inbound Submissions ... 71

12. RC Client Components ... 72

12.1 SFTP Client ... 72
12.2 Compression Utility ... 73
12.3 Encryption Utility ... 73
12.4 XML Processor ... 73
12.5 Scheduler ... 73
12.6 Housekeeping Manager ... 73

13. RC Client Workflow ... 74

13.1 Start RC Client ... 75
13.1.1 Login and Encryption ... 75
13.2 Outbound Process ... 75
13.2.1 Outbound Start ... 75
13.2.2 Get Outbound Documents ... 75
13.2.3 Connect ... 75
13.2.4 Push ... 75
13.3 Inbound Processes ... 76
13.3.1 Inbound Start ... 76
13.3.2 Housekeeping ... 76
13.3.3 Extraction ... 76

13.3.4 Checksum Verification ... 76
13.4 Acknowledgements ... 76
13.4.1 Pickup Notification ... 76
13.4.2 Error Pickup Notification ... 77
13.5 Connect ... 77
13.6 Get Notifications ... 77
13.7 Process Document ... 77
13.8 Pull Document ... 77

14. Release 4.0 Changes in the API ... 78

15. Java Client API ... 81

15.1 Security ... 81
15.2 Java API Documentation ... 82
15.2.1 Login ... 82
15.2.2 Inbound ... 83
15.2.3 Outbound ... 85
15.2.4 PA Review Decision Response ... 86
15.2.5 PA Error (Rejected Decision) Response ... 87
15.2.6 Administrative Error Response to Inbound Submissions ... 88
15.2.7 Utilities - Encryption ... 88
15.2.8 Test Connection ... 90
15.3 Logs ... 90
15.4 Utilities ... 91

16. Error Codes ... 92

16.1 Errors: esMD to RC ... 92
16.2 Errors: RC to esMD ... 93
16.2.1 Administrative Errors ... 94
16.2.2 Pickup Errors ... 94

17. Contacts ... 95

Appendix A: Description of Fields on RC Client Tabs ... 96

Appendix B: Reject Error Codes ... 99

Appendix C: Industry Codes ... 100

Appendix D: PA Reason Codes ... 101

Appendix E: PA Procedure Codes ... 102

Appendix F: Data Directories ... 104

Appendix G: Content Type Codes ... 106

Appendix H: Record of Changes ... 107

Appendix I: Acronyms 108

Appendix J: Glossary 109

Appendix K: Referenced Documents 110

Appendix L: Approvals 111

List of Figures

Figure 1: RC Client Inbound and Outbound Process 5

Figure 2: Keystore Password Encryption 51

Figure 3: Private Key Password Encryption 51

Figure 4: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 1 57

Figure 5: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 2 58

Figure 6: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 3 59

Figure 7: RC Client Components 72

Figure 8: RC Client Workflow 74

Figure 9: Encryption and Decryption Process 82

List of Tables

Table 1: Medicare Contractors, Responsibilities, and Contact Information ..2

Table 2: Libraries 44

Table 3: Keystore Creation Parameters 46

Table 4: JCE Policy Files 47

Table 5: Java Development Kit 48

Table 6: Java Runtime Environment 48

Table 7: Security Directory 48

Table 8: Sample RC Client Configuration File 49

Table 9: Inbound Files 53

Table 10: Outbound Files 54

Table 11: E_185457-flatfilerendering.ffr 56

Table 12: E_123456-Metadata.xml 59

Table 13: A_123456_Pickup_HIH_Status_Response.xml 61

Table 14: R_123456_ Pickup_Validation_Error.xml 61

Table 15: S_123456_Administrative_Error_HIH_Status_Response.xml 62

Table 16: M_123456_Administrative_Response_Validation_Error.xml 63

Table 17: X_123456_Virus_Scan_Error.xml 63

Table 18: N_123456_PA_Review_Result_HIH_Status_Response.xml 64

Table 19: V_123456_PA_Review_Response_Validation_Error.xml 64

Table 20: P_186303_Pickup_Notification.xml 65

Table 21: P_186303_Pickup_Error_Notification.xml 66

Table 22: E_1523121_PA_Review_Response.xml 66

Table 23: E_1523124_PA_Review_Response.xml 67

Table 24: E_1541233_PA_Review_Response.xml 68

Table 25: E_1521342_PA_Review_Response.xml 69

Table 26: D_1532432_Administrative_Error_Response.xml 71

Table 27: Inbound Client Method 78

Table 28: Outbound Client Method (PA Review Decision) 79

Table 29: Outbound Client Method (PA Error (Rejected Decision) Response) 80

Table 30: Outbound Client Methods (Administrative Error Response) 81

Table 31: Login Details 83

Table 32: Inbound Method Details 84

Table 33: Retrieval of Outbound Documents Details 86

Table 34: Manual Submission of PA Review Decision Response 86

Table 35: Manual Submission of PA Error (Rejected Decision) Response . 87

Table 36: Manual Submission of Administrative Error Response 88

Table 37: Encryption 89

Table 38: Remote Troubleshooting 90

Table 39: RC Client Logs 90

Table 40: RC Client Utilities 91

Table 41: Error Codes Sent From esMD to RC 92

Table 42: Administrative Error Codes 94

Table 43: Pickup Error Codes 94

Table 44: Support Points of Contact 95

Table 45: Descriptions of Fields on Review Decision Response to PA Request Tab 96

Table 46: Descriptions of Fields on Error Response to PA Request Tab.... 97

Table 47: Descriptions of Fields on Administrative Error Response to Inbound Submissions Tab 97

Table 48: Descriptions of Fields on Advanced/Debugging Tab 98

Table 49: Procedure Codes for the PA Programs 102

Table 50: Inbound File Names and Data Directories 104

Table 51: Outbound File Names and Data Directories 105

Table 52: Content Type Code Descriptions 106

Table 53: Content Type Codes and Business Types 106

Table 54: Record of Changes 107

Table 55: Acronyms 108

Table 56: Glossary 109

Table 57: Referenced Documents..... 110

1. Introduction

The Centers for Medicare & Medicaid Services (CMS) is a federal agency that ensures health care coverage for more than 100 million Americans. The CMS administers Medicare and provides funds and guidance for all of the 50 states in the nation, for their Medicaid programs and Children's Health Insurance Program (CHIP). The CMS works together with the CMS community and organizations in delivering improved and better coordinated care.

1.1 Overview of the esMD

Each year, the Medicare Fee-For-Service (FFS) Program makes billions of dollars in estimated improper payments. The CMS employs several types of Review Contractors (RCs) to measure, prevent, identify, and correct these improper payments. RCs find improper payments and manually review claims against medical documentation obtained to verify the providers' compliance with Medicare rules. The RCs request medical documentation by sending a paper letter to the provider. In the past, medical documentation providers had only two options for delivering the medical documentation requested by sending it by letter or fax.

The Electronic Submission of Medical Documentation (esMD) system gives providers the option of sending medical documentation electronically to a requesting RC, instead of sending the documentation by letter or fax.

Many providers use a Health Information Handler (HIH) organization to perform tasks, such as submitting claims and providing electronic health record systems. Any organization that handles health information on behalf of a provider is an HIH. Some HIHs are beginning to offer esMD gateway services; Claim Clearinghouses, Release of Information vendors, Health Information Exchanges, and Electronic Health Record vendors are often referred to as HIHS.

The esMD system allows providers and HIHs use gateway services to send responses for requests for additional documentation electronically to an RC during the claims review process.

1.1.1 The esMD Claim Review Contractors

Under the authority of the Social Security Act, CMS employs a variety of contractors to process and review claims in accordance with Medicare rules and regulations. Table 1: Medicare Contractors, Responsibilities, and Contact Information lists the review contractors referenced in this implementation guide.

Table 1: Medicare Contractors, Responsibilities, and Contact Information

Type of Contractor	Responsibilities	Contact Information
Medicare Administrative Contractors (MACs)	Process claims submitted by physicians, hospitals, and other health care professionals, and submit payment to those providers in accordance with Medicare rules and regulations. This includes identifying and correcting underpayments and overpayments.	http://www.cms.gov/ Research-Statistics-Data- and-Systems/Monitoring-Programs/Medicare-FFS-Compliance-Programs/Review-Contractor-Directory- Interactive-Map
Zone Program Integrity Contractors (ZPICs), formerly Program Safeguard Contractors (PSCs)	Identify cases of suspected fraud and take appropriate corrective actions.	http://www.cms.gov/ Research-Statistics-Data- and-Systems/Monitoring-Programs/Medicare-FFS-Compliance-Programs/Review-Contractor-Directory- Interactive-Map
Supplemental Medical Review Contractor (SMRC)	Conduct nationwide medical review, as directed by CMS. This includes identifying underpayments and overpayments.	http://www.cms.gov/Research-Statistics-Data-and-Systems/Monitoring-Programs/Medicare-FFS-Compliance-Programs/Medical-Review/SMRC.html
Contractor (CERT DC), CERT Review Contractor (CERT RC), and CERT Statistical Contractor (CERT SC)	Collect documentation and perform reviews on a statistically-valid random sample of Medicare FFS claims to produce an annual improper payment rate.	https://www.cms.gov/Research-Statistics-Data-and-Systems/Monitoring-Programs/Medicare-FFS-Compliance-Programs/CERT/index.html?redirect=/cert
Recovery Auditors	Identify underpayments and overpayments, as part of the Recovery Audit Program.	http://www.cms.gov/Research-Statistics-Data-and-Systems/Monitoring-Programs/Medicare-FFS-Compliance-Programs/Recovery-Audit-Program/

1.2 System Overview

The esMD system provides a mechanism for exchanging medical documentation and responses for Cross-Enterprise Document Reliable Interchange (XDR) and X12N 278 requests between the Medicare Provider community and the Medicare RC community. The purpose is to enable the electronic transmission of information between HIHs who represent Providers and the Medicare RCs, replacing paper documents where possible.

The RC Client is a utility that enables RCs to communicate with esMD by exchanging files via TIBCO® Managed File Transfer (MFT) server.

Note: The esMD system identifies submissions and requests sent from HIHs to RCs as inbound files, and identifies transactions and responses for XDR and X12N 278 sent from RCs to HIHs as outbound files.

1.3 System Requirements

See Section 8, System Requirements for the system requirements for installing a Microsoft Java version of the RC Client.

Section 8, System Requirements provides the requirements needed for the computer system where the RC Client will be installed, including the computer system's processor, amount of disk space and free memory needed, permissions, minimum internet connectivity Kilobits Per Second (Kbps) transfer speeds, and the Microsoft Java Framework version needed to run the RC Client properly.

Refer to the EIDM Instructions in the link below on details on how to obtain an EIDM login: http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Downloads/EIDMInstructions.docx

Refer to Section 1.4.3 RC Client Operation Overview for Enterprise File Transfer (EFT) Password requirements per CMS policy for logging in to the internal server.

1.4 RC Client Overview

Since September 2011, the CMS has implemented the esMD program for providers to submit medical documentation in response to requests from Medicare RCs and also enhanced the esMD Gateway to support electronic responses to requests.

In September 2012, the CMS implemented a Prior Authorization (PA) process via the esMD Gateway for Power Mobility Devices (PMD) for FFS Medicare beneficiaries who reside in seven states with high populations of error prone providers (California, Florida, Illinois, Michigan, New York, North Carolina, and Texas).

In January 2013, CMS expanded the CMS esMD Gateway to allow Durable Medical Equipment (DME) suppliers and providers to send electronic PA Requests to Medicare RCs.

In June 2013, CMS enabled automated "Prior Authorization Review Results Responses" from Medicare RCs to HIHs via the esMD Gateway.

In June 2014, the "RC Client" application was implemented to allow data exchanges between HIHs and facilitate Medicare RCs electronically receiving PA requests to the RC's computer system and allow RCs to electronically enter a decision on a PA request.

In June 2015, the "RC Client" application was extended to allow RCs to enter a Reject Error

Code for a PA request electronically received, or electronically submit that there was an error in receiving the PA request's response that was transmitted to the RC Client installed on the Medicare RC's computer system or network. The RC will be able to submit responses for PA programs, such as the Ambulance and Hyperbaric Oxygen (HBO) and PMD programs.

In July 2016, the "RC Client" application will receive Durable Medical Equipment, Prosthetics, Orthotics and Supplies (DMEPOS) PA request and Home Health Services(HHS) PA request and will be able to send PA responses for these PA programs. The "RC Client" application will also receive Second Level Appeal Requests via the esMD system.

1.4.1 RC Client Pull/Push Functionality

The RC Client provides the following functionality:

- Pull:
  - Inbound documents (submitted by HIHs) from the TIBCO MFT server;
  - HIH acknowledgements indicating receipt of pick up notifications and PA review result responses; and
  - Data Element Validation results for the outbound process.
- Push:
  - PA review decision responses to PA Requests for XDR and X12N 278 to esMD;
  - Error responses to PA Requests for XDR and X12N 278 to esMD;
  - Administrative Error response for XDR and X12N 278 to esMD;
  - Error messages generated due to file decompression and checksum verification;
  - Acknowledgement messages for receipt of documents and authorization requests; and
  - Site-Specific Configuration settings:
    - Push frequency/Pull frequency; and
    - Folder locations for both Inbound and Outbound files.

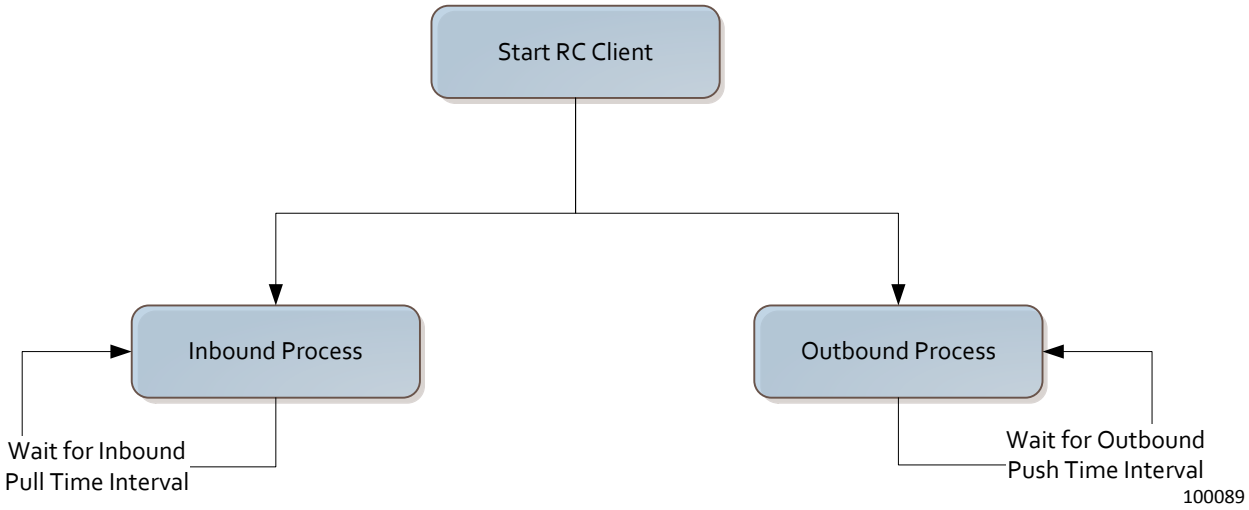
1.4.2 RC Client Application Overview

The esMD RC Java Client is a standalone Java Windows desktop application that runs outside the CMS network on the RC's machine, computer, or server. The purpose of the RC Java Client is to connect to the TIBCO MFT server at the Baltimore Data Center (BDC) and push and pull files. The RC Java Client uses the Enterprise Identity Management (EIDM) System login credentials to authenticate with the TIBCO MFT server. The RC Client users (at the RC site) provide their login credentials when they start the RC Client on their machines.

Users enter their login credentials only once at the program startup. When the RC Client starts, it initiates and then continuously runs two parallel threads as shown in Figure 1: RC Client Inbound and Outbound Process. When a user starts the RC Client, it will run continuously and

will push and pull files automatically without continual user intervention, based on the frequencies set by the RC.

Figure 1: RC Client Inbound and Outbound Process



In the inbound process, when the RC Client connects to the TIBCO MFT server, the RC Client immediately executes a pull cycle. The documents are pulled into the RC's inbound user directory for the authenticated user, and then the RC Client disconnects and waits for the next cycle, as determined by the Inbound Pull Time Interval setting.

In the outbound process when the RC Client connects to the TIBCO MFT server, the RC Client executes a push cycle. The documents are pushed from the RC's outbound user directory to the TIBCO MFT server, and then the RC Client disconnects and waits for the next cycle as determined by the Outbound Push Time Interval setting.

The inbound pull frequency is independent of the outbound push frequency. After each successful push or pull process, the RC Client thread disconnects from the TIBCO MFT server. To ensure continuous operation of the RC Client, it must preserve each user's EIDM login credentials during the program execution.

Note: Running multiple instances of the Java RC Client for the same jurisdiction could result in errors while pulling the files.

The RC Client has been updated as part of esMD Release 4.0 to allow RCs to submit review responses for the new PA programs using the Graphical User Interface (GUI). The RC does not need to login to the TIBCO MFT Server in order to create Review Responses, Error Responses and Administrative Error Responses. The login is necessary only to pull or push files from or to TIBCO MFT Server.

1.4.3 RC Client Operation Overview

The RC Client runs in a cyclical manner sleeping for a specified time interval between the

operating cycles. The sleep intervals are configured in the "checkFrequency" parameter for the Inbound Process and the "pushFrequency" parameter for the Outbound process. The RC is advised to use the default of 240 minutes (4 hours) for the Inbound process and 15 minutes for the Outbound process.

The RC Client operation is interrupted in two events:

1. EIDM passwords that have expired (Note: EIDM passwords expire every 60 days, if not changed); and
2. A Virus Scan error notification is received from the esMD.

In the first scenario, when the EIDM password expires, the RC Client suspends its operation and is terminated. The RC must restart the RC Client and the user must provide the right credentials to login to the TIBCO MFT server. The EIDM notifies the user 15 days prior to the password expiring. For more information on the EIDM User Credentials and how to reset the password, please refer to the EIDM Instructions document in the esMD Downloads section, using the link below:
http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Information_for_Review-Contractors.html

The password setup in the portal must meet the following CMS policy (which differs from EIDM's policy) for users to be able to log into Internet Server:

PASSWORD POLICY

1. Check password against the dictionary
2. Maximum Length: 8
3. Maximum Special: 0
4. Minimum Alpha: 1
5. Minimum Begin Alpha: 1
6. Minimum Length: 8
7. Minimum Lowercase: 1
8. Minimum Number of Character Type Rules That Must Pass: All
9. Minimum Numeric: 1
10. Minimum Uppercase: 1
11. Must not contain values of attributes: accountId
12. Must not contain words: 1234, PASSWORD, WELCOME, CMS, HCFA, SYSTEM, MEDICARE, MEDICAID, TEMP, LETMEIN, GOD, SEX, MONEY, QUEST, F20ASYA, RAVENS, REDSKIN, ORIOLES, BULLETS, CAPITOL, MARYLAND, TERPS, DOCTOR, 567890, 12345678, ROOT, BOSSMAN, JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER, SSA, FIREWALL, CITIC, ADMIN, UNISYS, PWD, SECURITY, 76543210, 43210, 098765, IRAQ, OIS, TMG, INTERNET, INTRANET, EXTRANET, ATT, LOCKHEED

Note: After the password reset, update the password to the new password in the configuration or script file if it is being stored and used by RC Client.

In the second scenario, when a Virus Scan error notification has been received from esMD, all the processes of the RC Client are suspended and the RC Client is terminated. In addition, the RC Client is locked and cannot pull/push files even if the RC Client is restarted. The RC is advised to contact the esMD helpdesk (refer to Section 17. Contacts for more details) to unlock the RC Client.

2. Overview of How This Document is Structured

This document is structured into the following two primary sections.

1. First primary section of this document provides the following:

   - How to start and log into the RC Client;
   - How to enter a Review Response decision;
   - How to enter an error code for a PA request;
   - How to submit Inbound Submissions errors; and
   - Advanced debugging, which shows how to test to see if your RC Client application can connect to the TIBCO MFT server and if you have any inbound files ready for downloading.

   ❖ The audience for this first section is intended for the **RC business users**.

2. How to install and configure a Java version of RC Client.

   ❖ The audience for this second section is intended for the **person(s) installing the RC Client application**.

   This section provides the technical specifications for installing and configuring RC Client on a computer system or network and includes the following:

   - Overview of the installation process;
   - Systems Requirements for a Java installation;
   - Installing an Out-of-Box Java version of the RC Client application;
   - TIBCO MFT file transfers;
   - XML Messages, including Outbound, Inbound, and Error messages;
   - Inbound Processes and Files;
   - Outbound Processes and Files;
   - Configuring the RC Client application;
   - RC Client Components;
   - RC Client Workflow;
   - RC Client application Utilities, Components, Schedulers, and Encryption;
   - Release 4.0 Changes to the Application Programming Interface (API);

- Using API;
- Configuring the RC Client application for notifications;
- Processing and pulling in documents; and
- Security.

3. How to Start the RC Client and Log In

The following are the step-by-step instructions for starting the RC Client and logging in.

Step 1.

Starting the RC Client and Logging In

Start the RC Client by selecting the **rcclient.bat** in the RC Installation folder or directory.

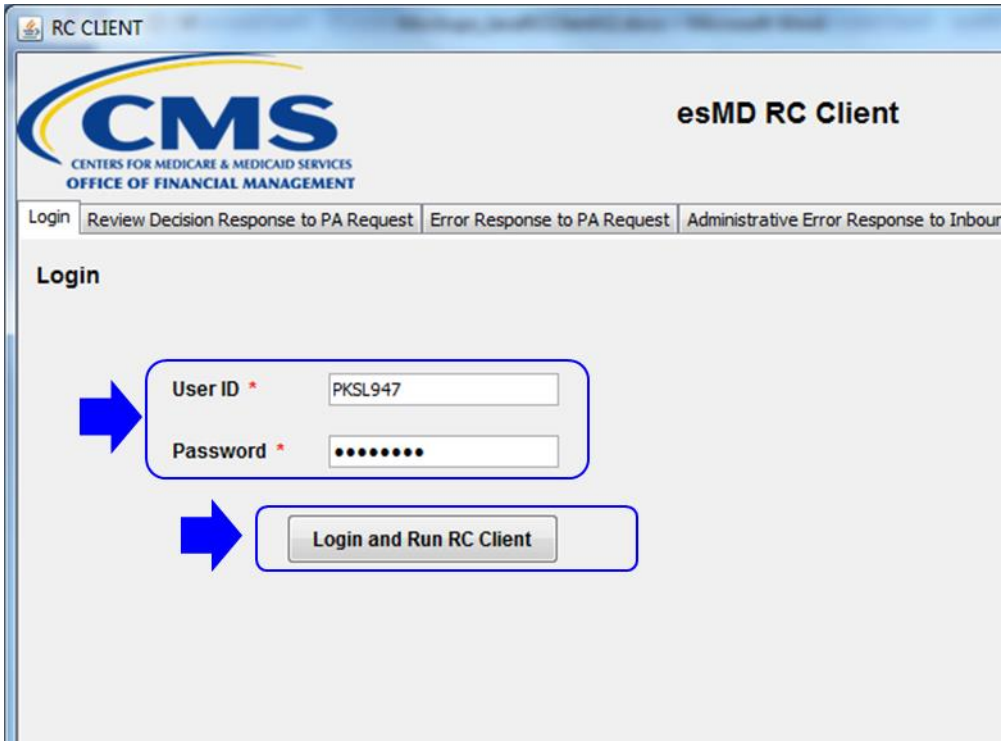
Step 2.

Starting the RC Client and Logging In

The **Login** screen is displayed.

Enter your EIDM **User ID** and **password** and then select **Login and Run RC Client**.

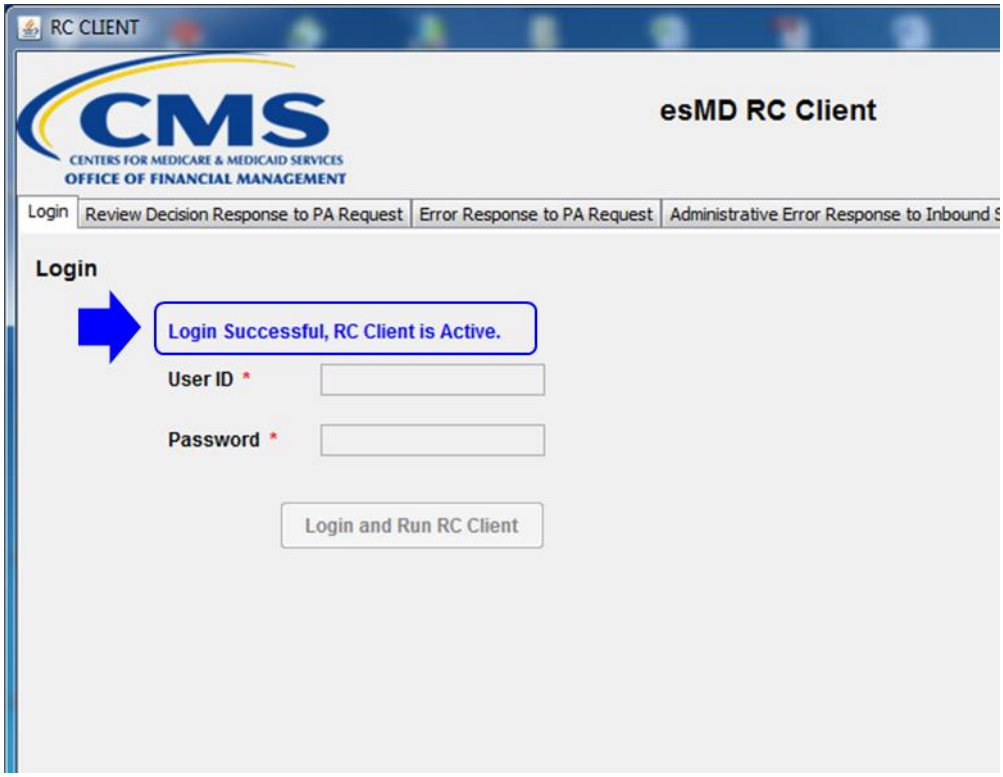
Note: The EIDM login credentials are confidential and should not be shared with others. (For more information on EIDM login credentials, see EIDM's Frequently Asked Questions (FAQs) https://portal.cms.gov/wps/portal/unauthportal/faq.



Step 3.

Starting the RC Client and Logging In

After a successful log in, the **Login Successful, RC Client is Active.** message is displayed.



4. How to Enter a Decision on the Review Decision Response to Prior Authorization (PA) Request Tab

This section provides step-by-step instructions on how to enter a decision on the **Review Decision Response to PA Request** tab for the following:

- How to Enter an A-Affirmed Decision
- How to Enter a M-Affirmed with Change Decision
- How to Enter a N-Non Affirmed Decision

Note: An RC has up to ten business days to process and respond to a PA Request.

4.1 How to Enter an A-Affirmed Decision

This section provides step-by-step instructions on how to enter an Affirmed decision on the **Review Decision Response to PA Request** tab.

Step 1.

Entering an **A-Affirmed** Decision

Select the **Review Decision Response to PA Request** tab.

❖ After a successful log in, another log in is not required to navigate to and use the Review Decision Response to PA Request tab.

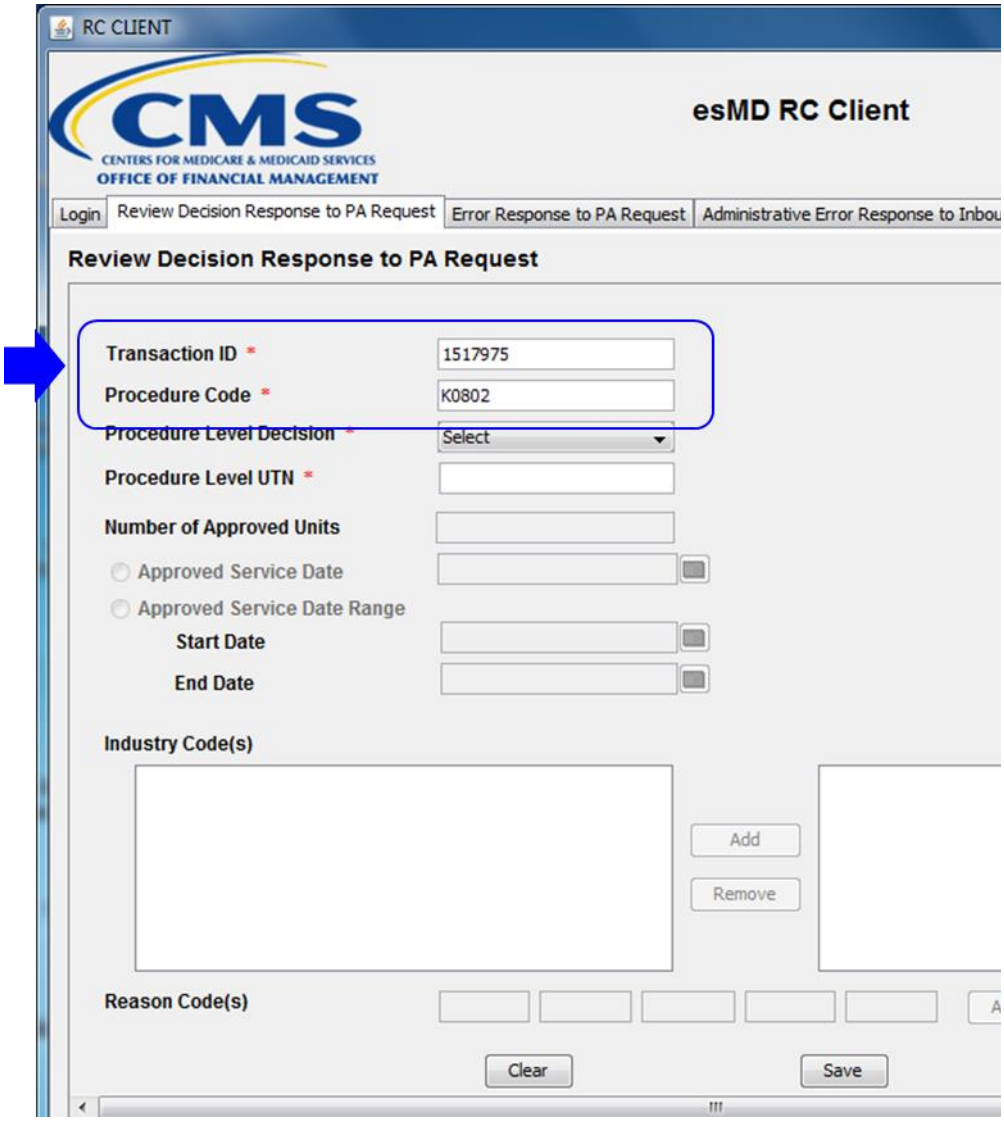
Step 2.

Entering an **A-Affirmed** Decision

The fields for the **Review Decision Response to PA Request** tab are displayed.

❖ Before You Begin: If you need a brief description of any of the fields on the tabs, see Appendix A: Description of Fields on RC Client Tabs on page 96.

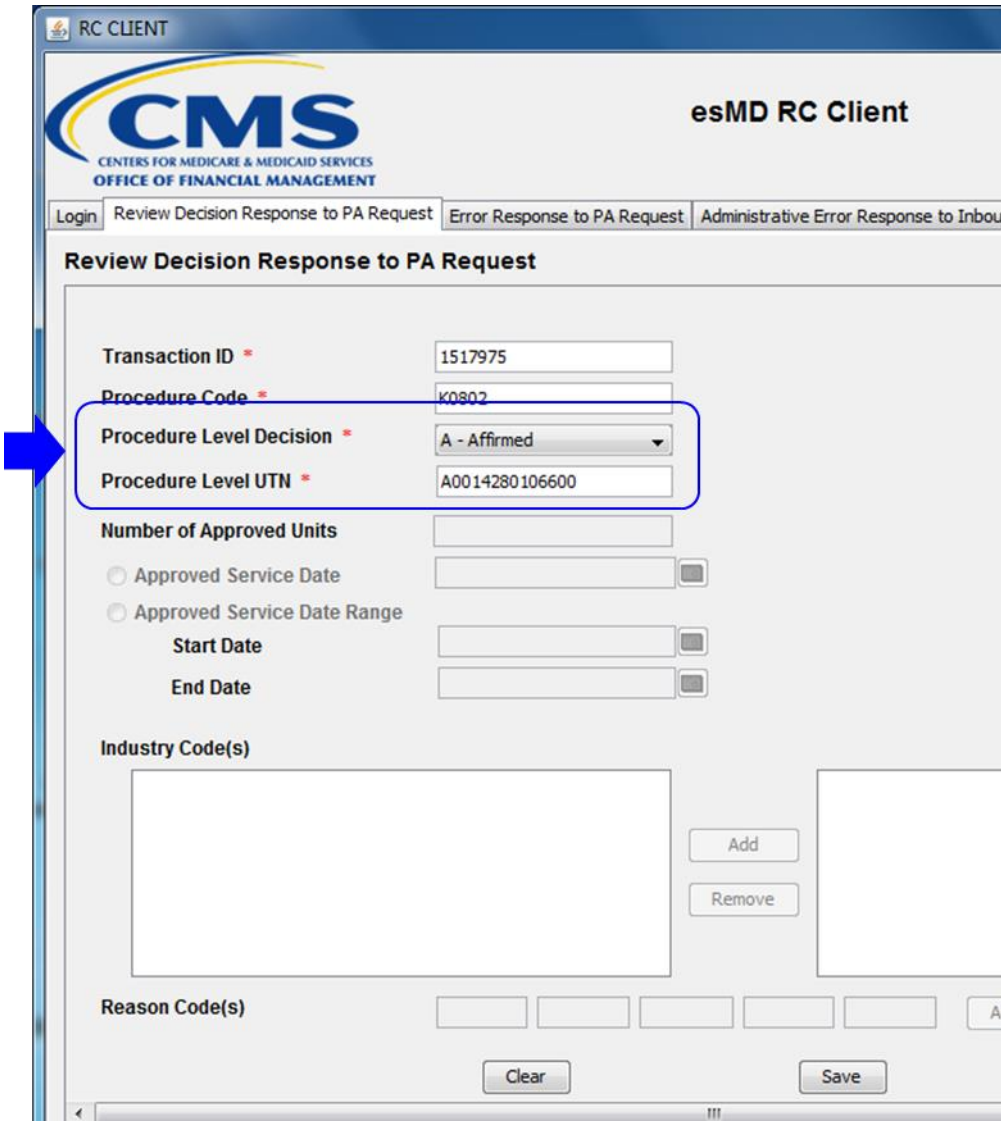
Enter the **Transaction ID** and the **Procedure Code**.



Step 3.

Entering an **A-Affirmed** Decision

Select the **A-Affirmed** decision from the **Procedure Level Decision** drop down menu and enter the **Procedure Level UTN**.

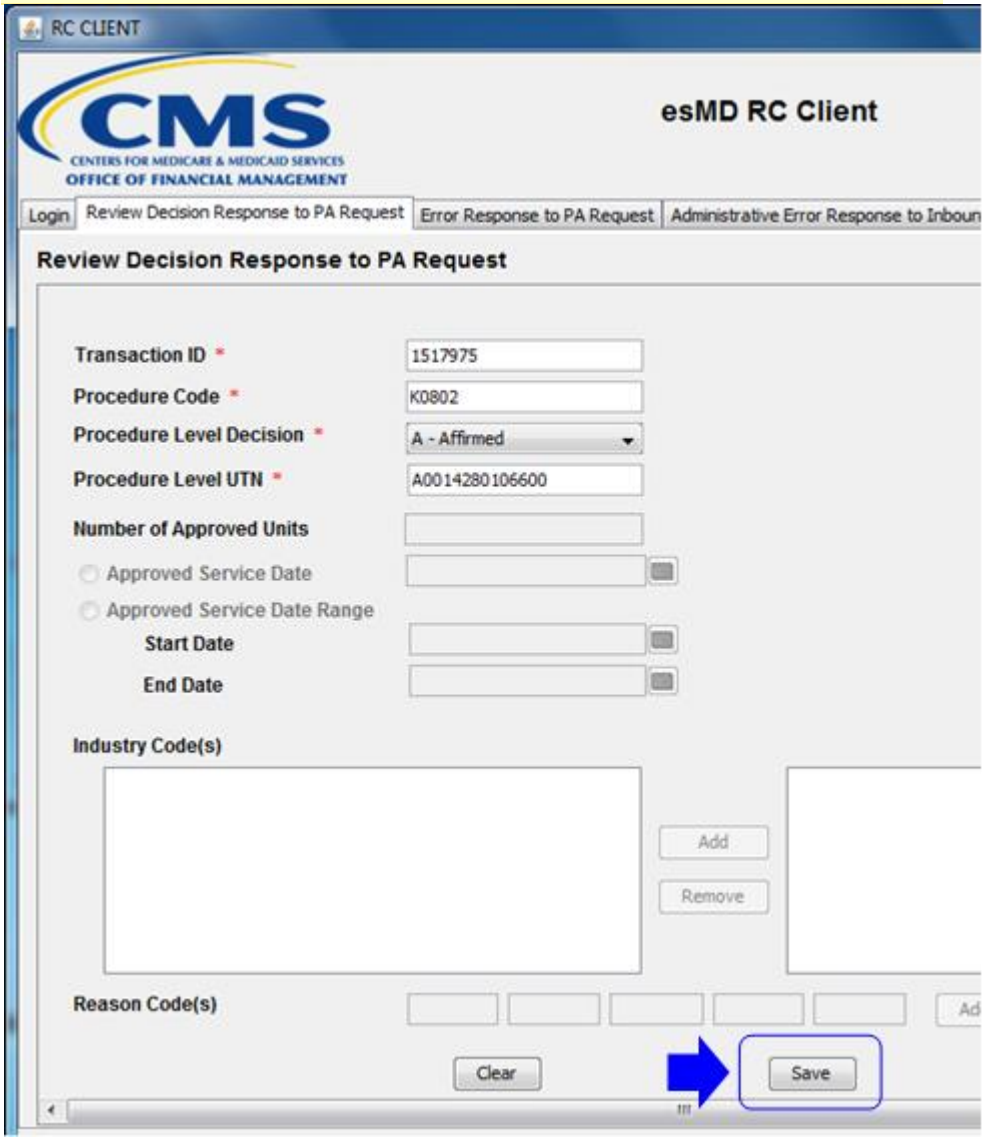


Step 4.

Entering an **A-Affirmed** Decision

Select **Save** to save the A-Affirmed decision for submission.

Technical Note: After selecting Save, an Extensible Markup Language (XML) message will be created to be sent to the esMD system and will be packaged into a compressed zip file. The zip file will be placed in a directory specified in the OutboundConfig/outputDirectory of the esmd-rc-client-config.xml. The outbound thread running on the RC Client will push the file to the TIBCO Managed File Transfer (MFT) server.



Step 5.

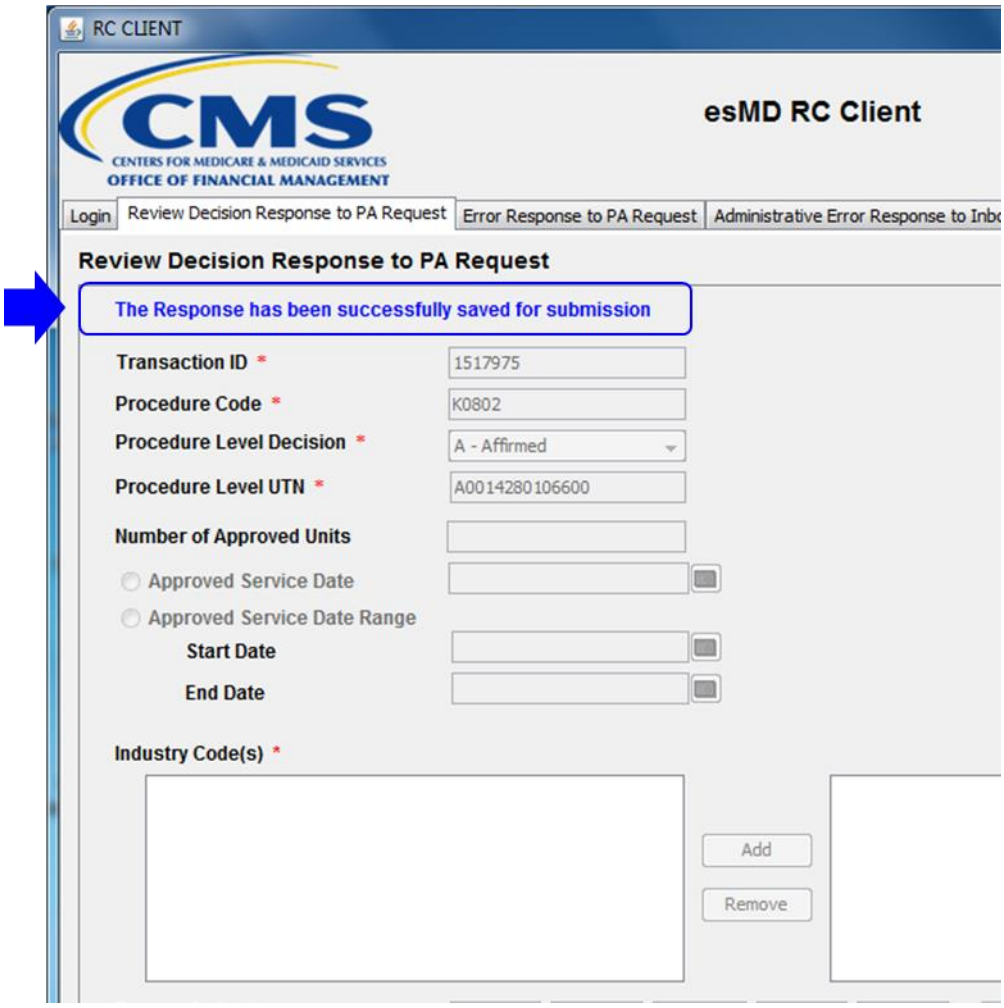
Entering an **A-Affirmed** Decision

After selecting Save, the "**The Response has been successfully saved for submission**" message is displayed.

Technical Note: After selecting Save, the RC Client validates the data entered and displays errors messages, as applicable.

If the data validation is successful, the A-Affirmed decision is created, and the "The Response has been successfully saved for submission" message is displayed.

Note: After successfully saving a decision for submission, all information in the fields are cleared and another response may be entered.



4.2 How to Enter a M-Affirmed with Change Decision

This section provides step-by-step instructions on how to enter an M-Affirmed with Change decision on the **Review Decision Response to PA Request** tab.

Step 1.

Entering a **M-Affirmed with Change** Decision

Select the **Review Decision Response to PA Request** tab.

❖ After a successful log in, another log in is not required to navigate to and use the Review Decision Response to PA Request tab.

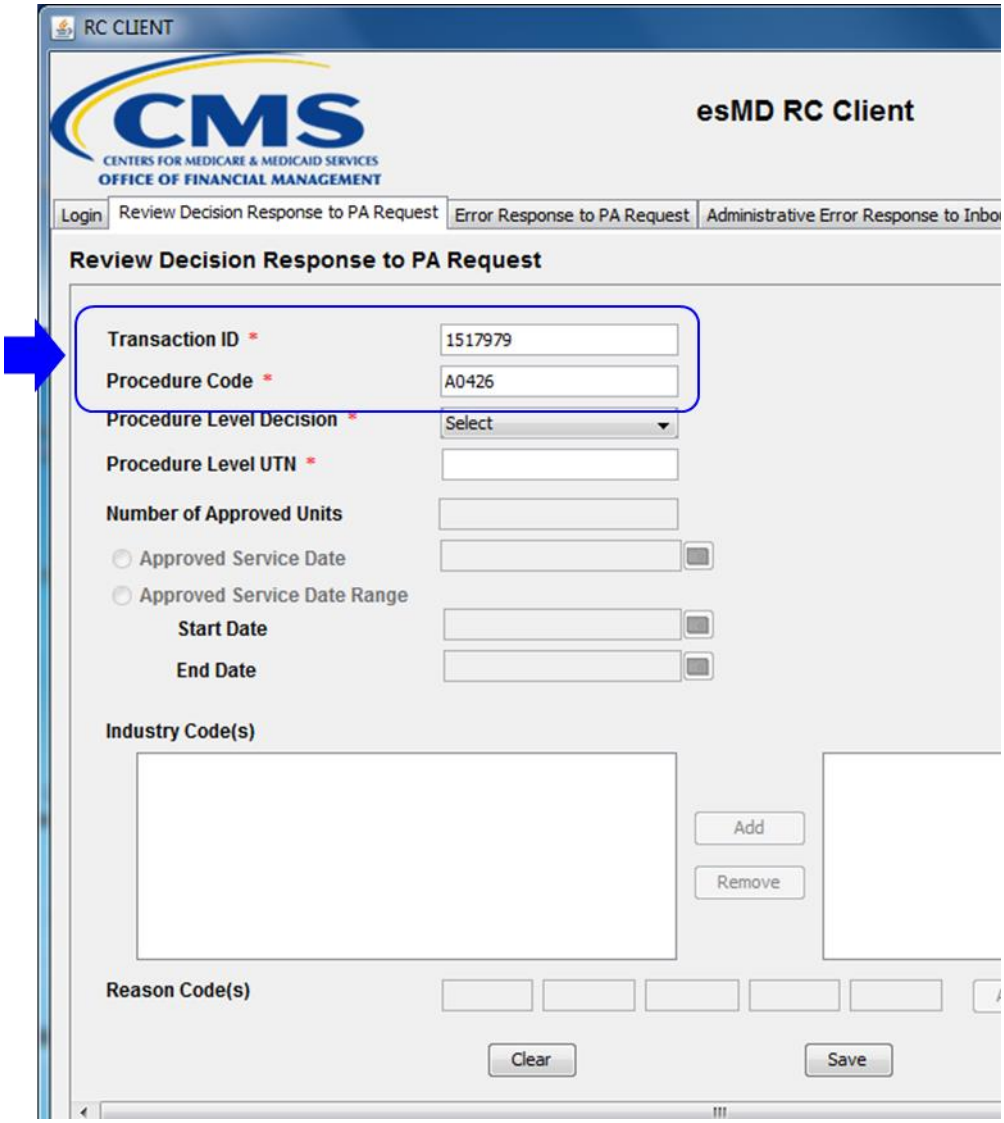
Step 2.

Entering a **M-Affirmed with Change** Decision

The fields for the **Review Decision Response to PA Request** tab are displayed.

❖ Before You Begin: If you need a brief description of any of the fields on the tabs, see Appendix A: Description of Fields on RC Client Tabs on page 96.

Enter the **Transaction ID** and the **Procedure Code**.



Step 3.

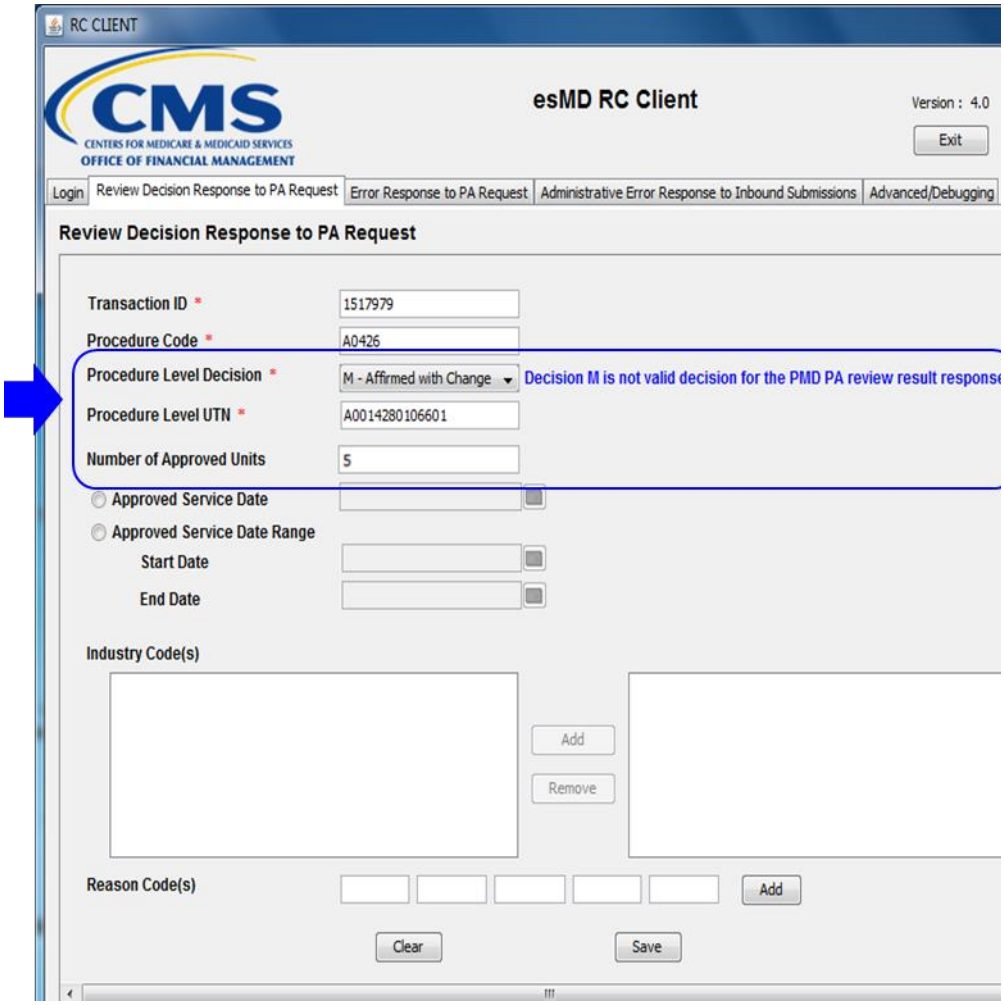
Entering a **M-Affirmed with Change** Decision

Select the **M-Affirmed with Change** decision from the **Procedure Level Decision** drop down menu.

Note: After selecting an M-Affirmed **with Change** decision, the **"Decision M is not valid decision for the PMD PA review result response**" message is displayed. An **M-Affirmed with Change** decision selected and saved for submission for a PMD PA review result response will be treated as an invalid response in the esMD system.

Enter the **Procedure Level Universal Tracking Number (UTN)**.

Enter the **Number of Approved Units**, when you also have to submit this information. (This is not a required field.)

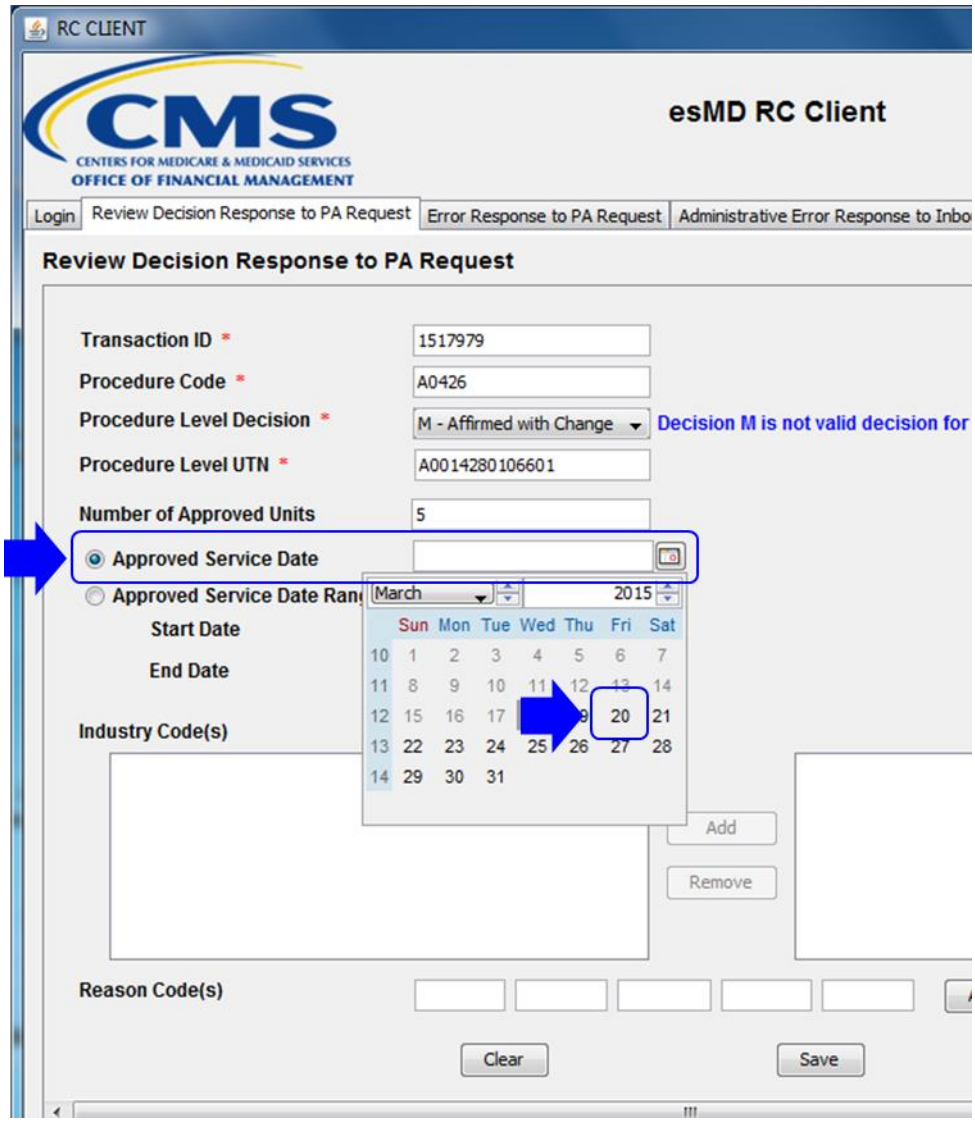


Step 4a.

Entering a **M-Affirmed with Change** Decision

Enter a single **Approved Service Date** or an **Approved Service Date Range**.

To enter a single **Approved Service Date**, select the date on the Approved Service Date calendar, as illustrated below.

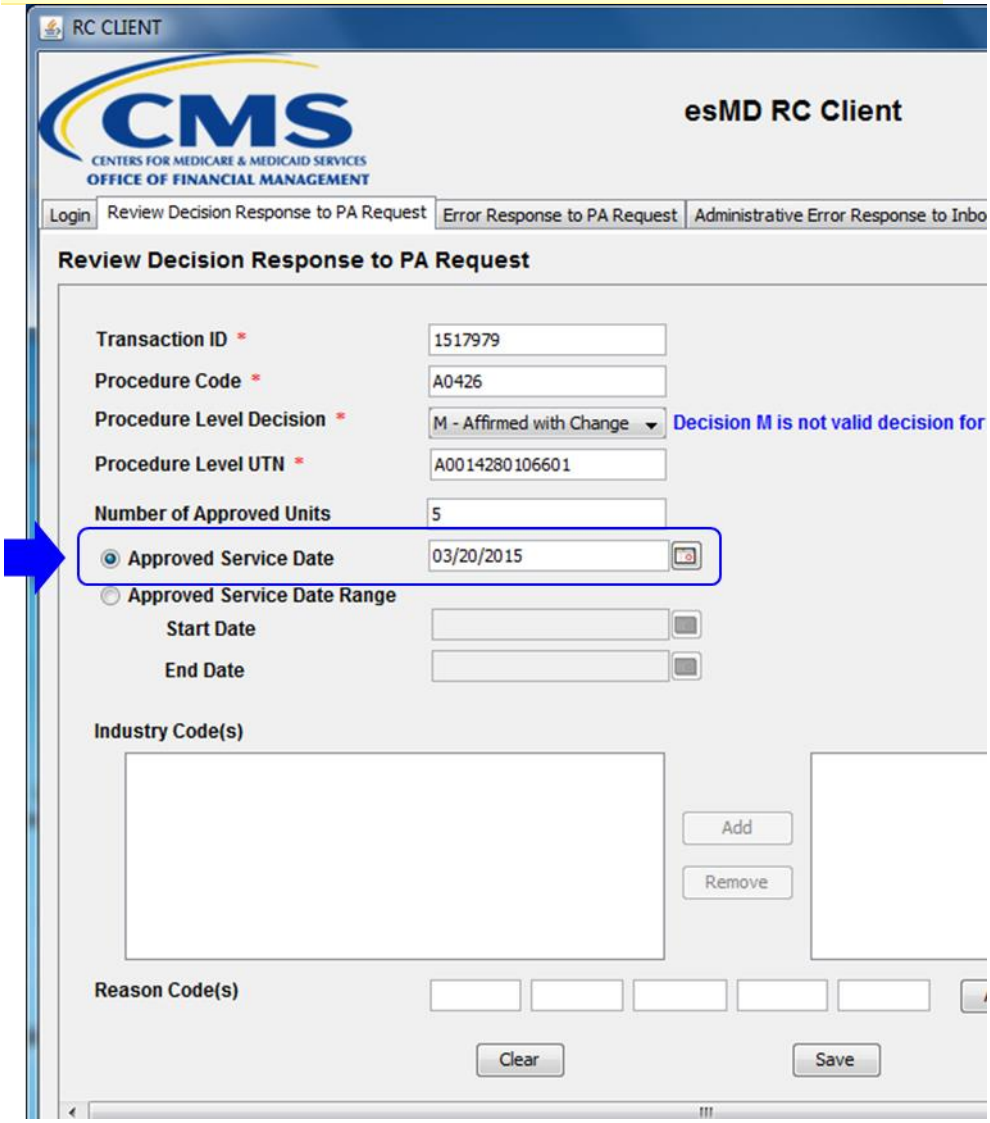


Step 4b.

Entering a **M-Affirmed with Change** Decision

The **date selected** is then placed in the Approved Service Date field.

❖ To enter an **Approved Service Date Range**, see Step 5.

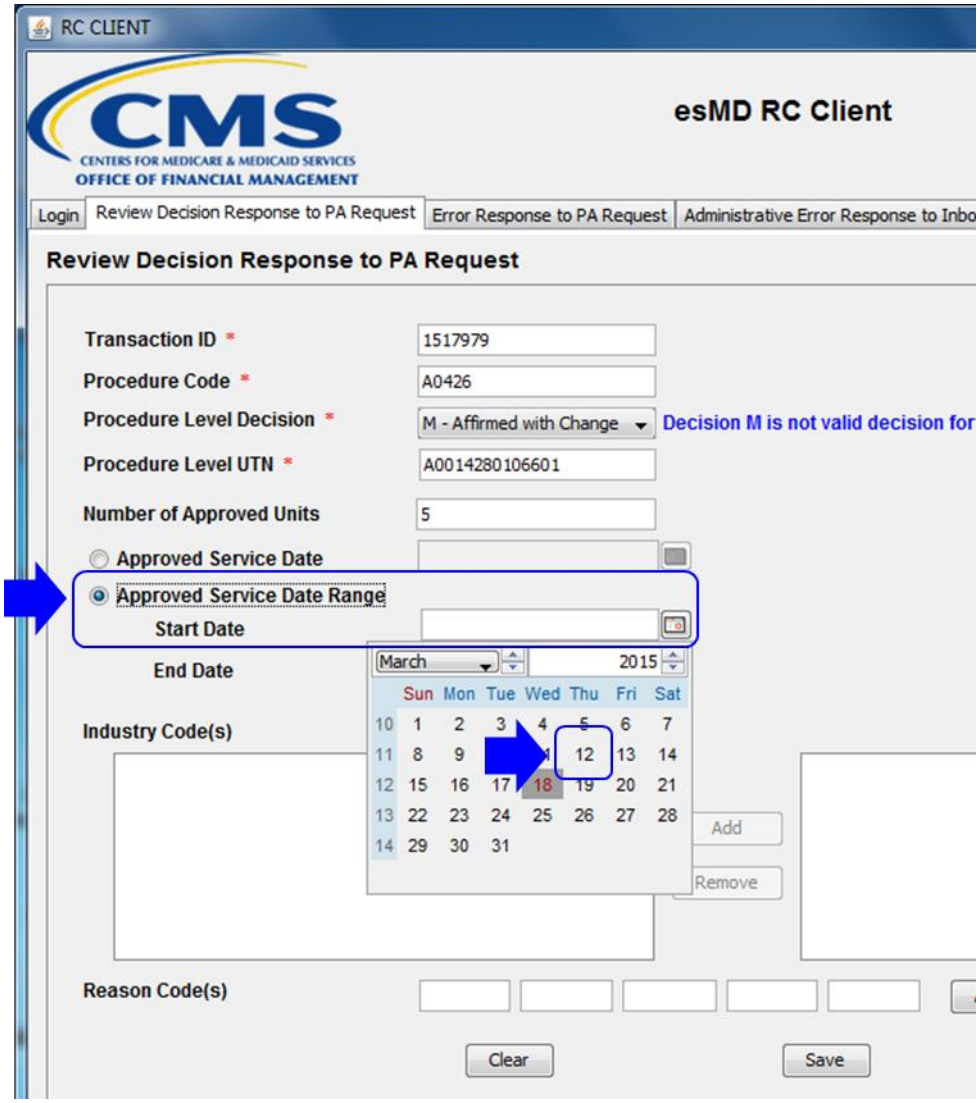


Step 5.

Entering a **M-Affirmed with Change** Decision

To enter an **Approved Service Date Range** (i.e. a Start Date and an End Date), first, select the date on the Approved Service Date calendar for the **Start Date**, as illustrated below.

Note: An RC has to provide either **Approved Service Date** or **Approved Service Date Range** as they are mutually exclusively.

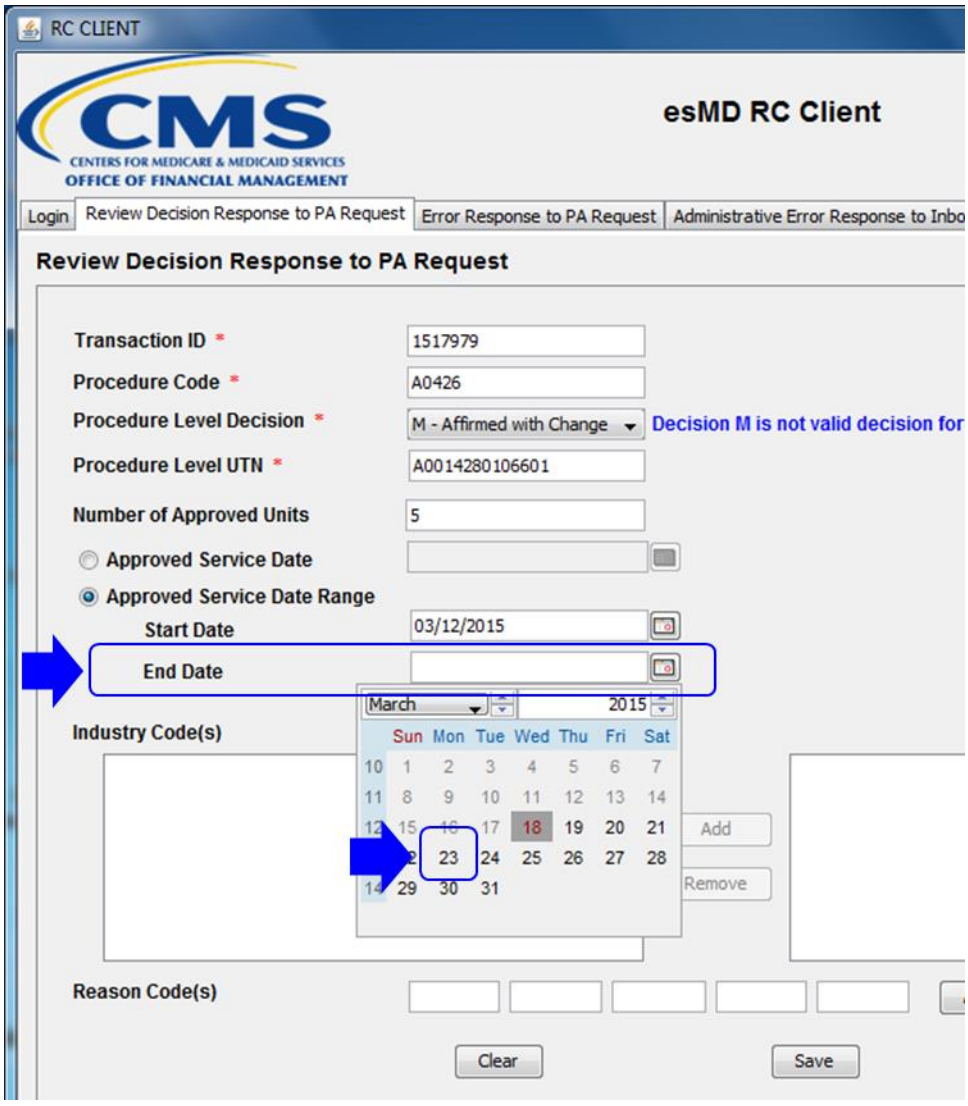


Step 5a.

Entering a **M-Affirmed with Change** Decision

Now, select the **date** on the Approved Service Date calendar for the **End Date**.

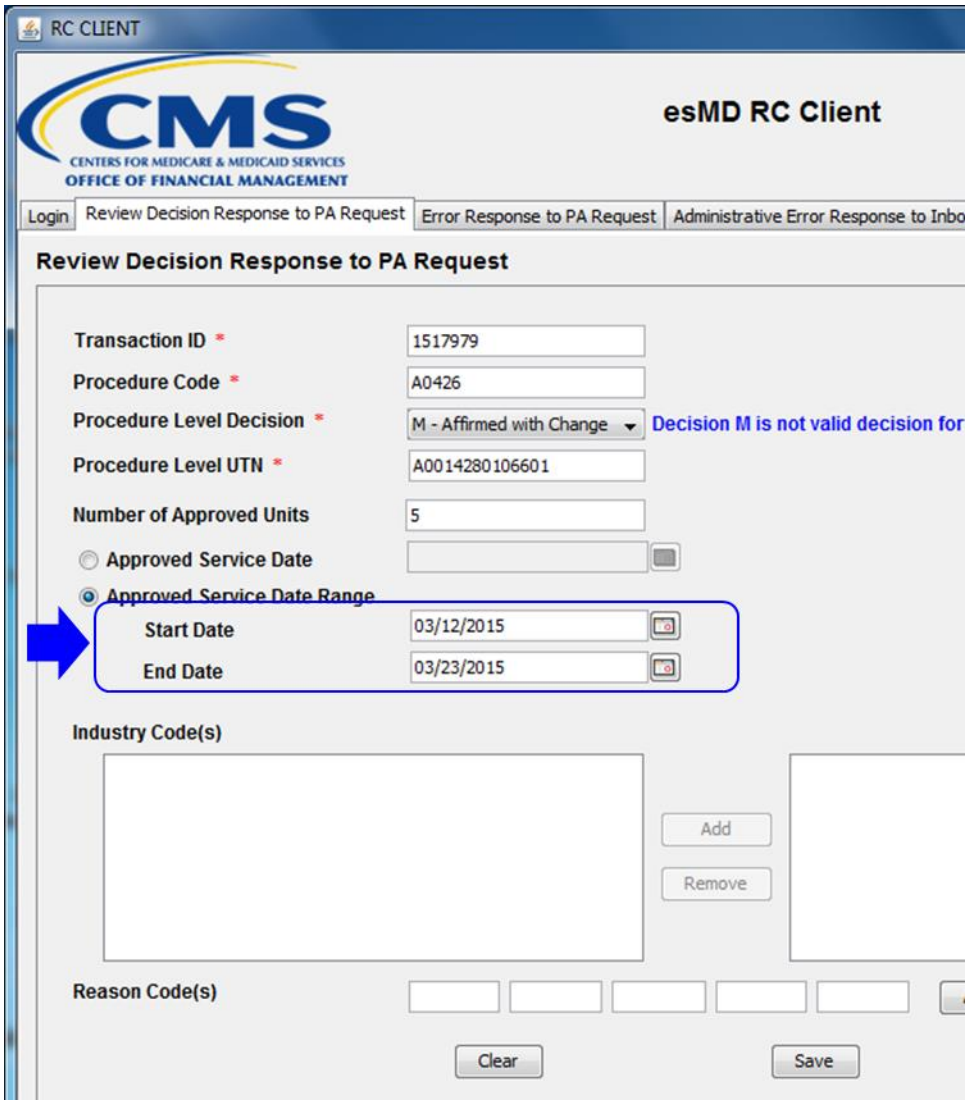
Note: The End Date cannot be the same date as the Start Date.



Step 5b.

Entering a **M-Affirmed with Change** Decision

The **Start Date** and **End Date** are added to the Start Date and End Date fields, as illustrated below.



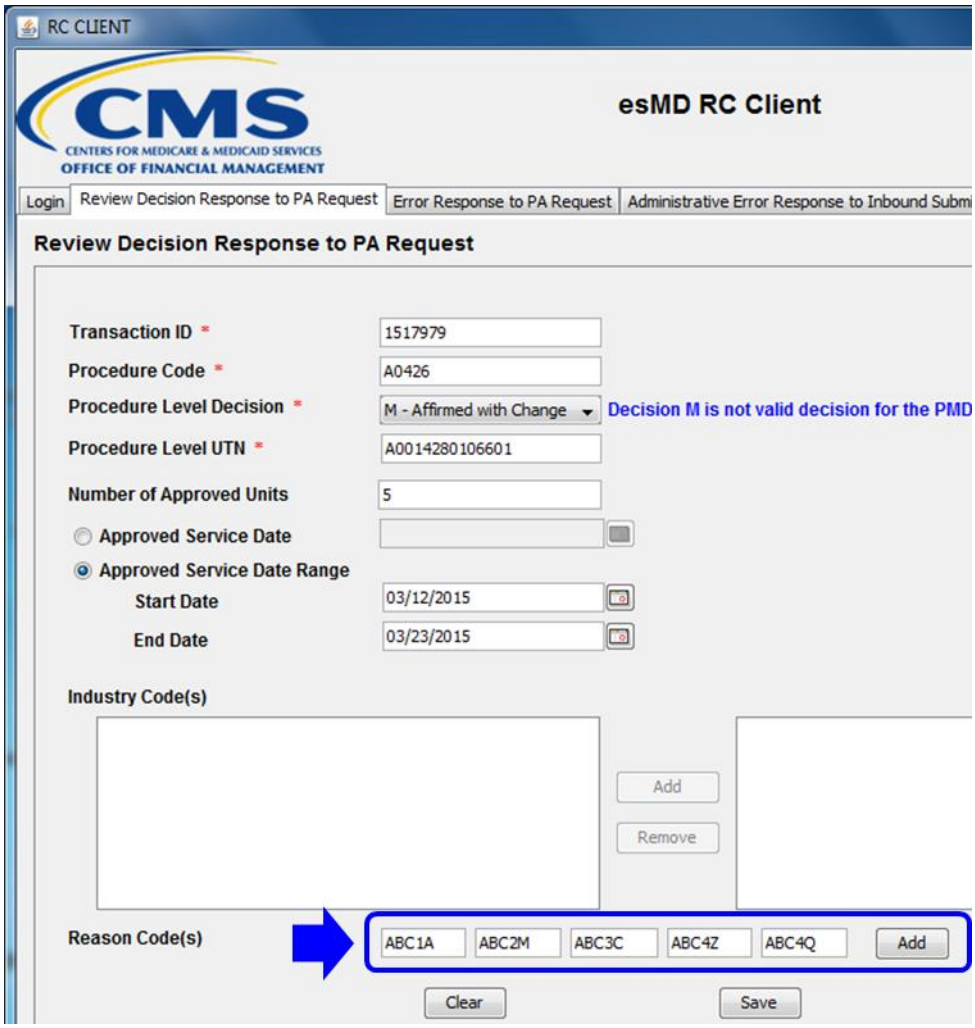
Step 6.

Entering a **M-Affirmed with Change** Decision

Enter a **Reason Code** or multiple **Reason Codes**.

For information on how to access an up-to-date list of PA Reason Codes, see Appendix D: PA Reason Codes.

Note: When you have to enter more than five Reason Codes, select **Add** at the end of the row of Reason Code fields to add a new row. Continue to select Add at the end of each row, until all of your Reason Codes have been entered. You may enter a maximum of 25 Reason Codes.

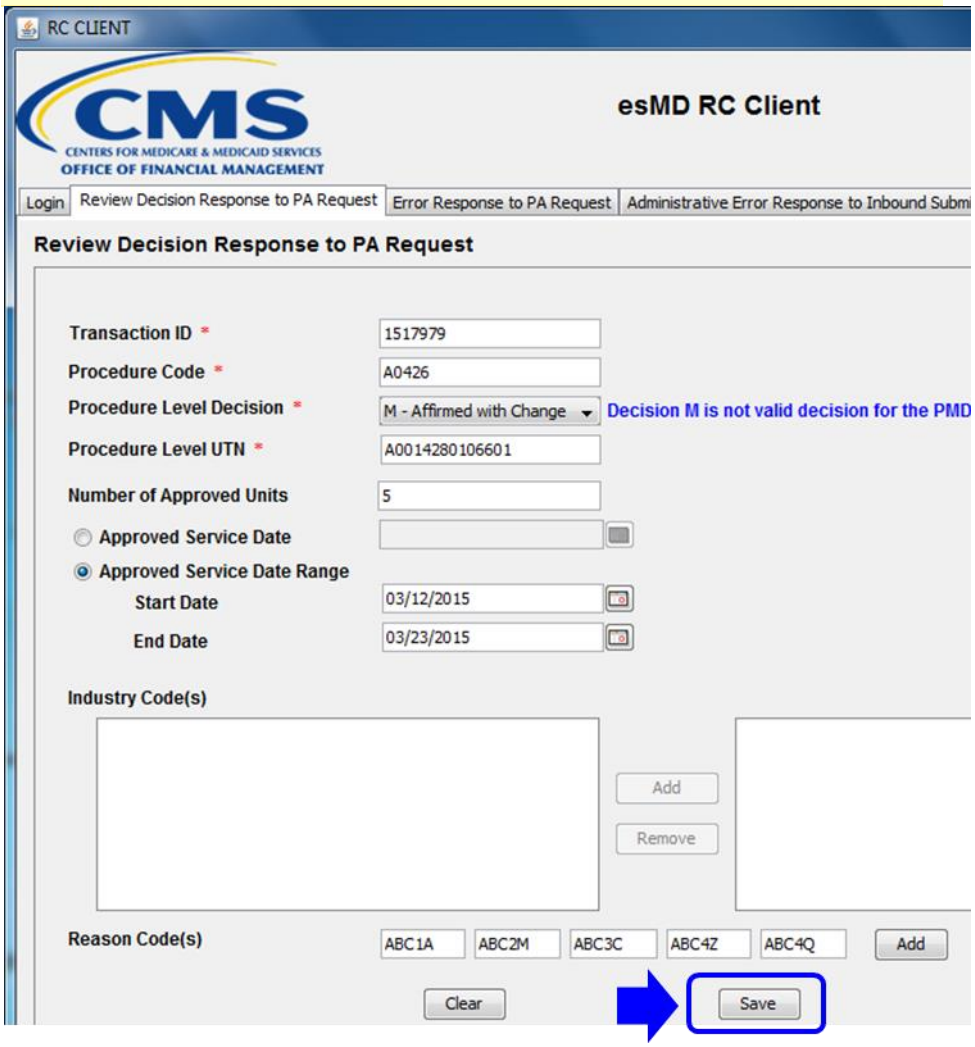


Step 7.

Entering a **M-Affirmed with Change** Decision

Select **Save** to save the M-Affirmed with Change decision for submission.

Technical Note: After selecting Save,, an Extensible Markup Language (XML) message will be created to be sent to the esMD system and will be packaged into a compressed zip file. The zip file will be placed in a directory specified in the OutboundConfig/outputDirectory of the esmd-rc-client-config.xml. The outbound thread running on the RC Client will push the file to the TIBCO Managed File Transfer (MFT) server.



Step 8.

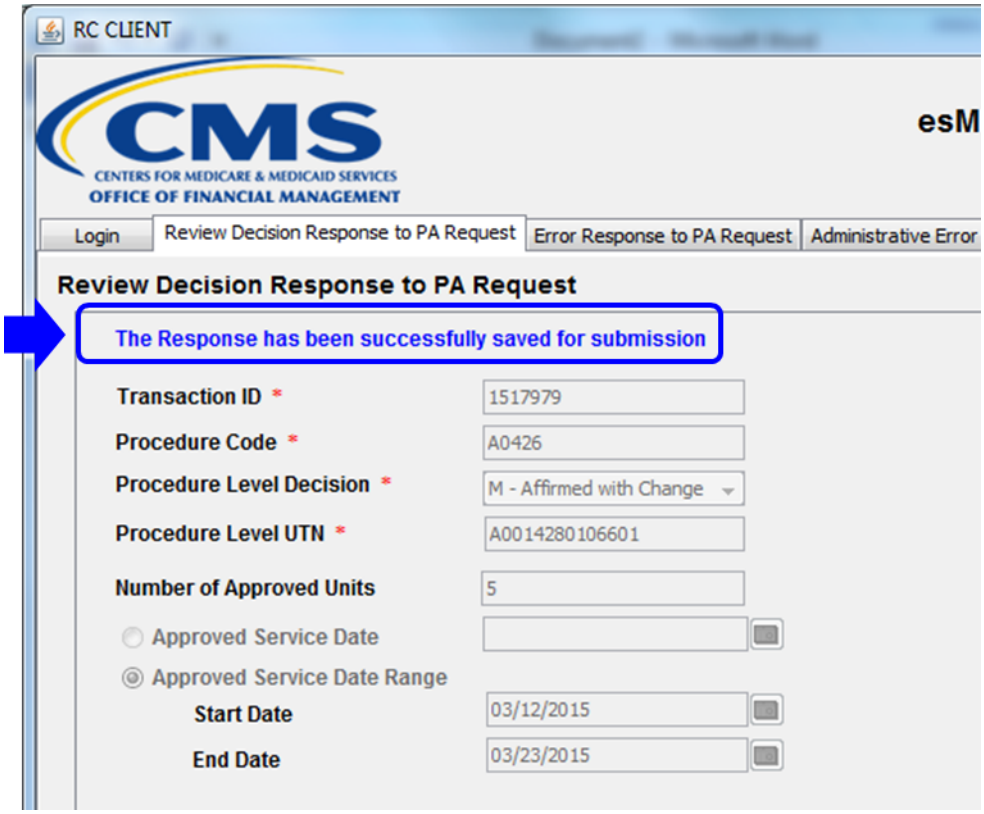
Entering a **M-Affirmed with Change** Decision

After selecting Save, the "**The Response has been successfully saved for submission**" message is displayed.

Technical Note: After selecting Save, the RC Client validates the data entered and displays errors messages, as applicable.

If the data validation is successful, the M-Affirmed with Change decision is created, and the "The Response has been successfully saved for submission" message is displayed.

Note: After successfully saving a decision for submission, all information in the fields are cleared and another response may be entered.



4.3 How to Enter a N-Non Affirmed Decision

This section provides step-by-step instructions on how to enter an N-Non Affirmed decision on the **Review Decision Response to PA Request** tab.

Step 1.

Entering a **N-Non Affirmed** Decision

Select the **Review Decision Response to PA Request** tab.

❖ After a successful log in, another log in is not required to navigate to and use the Review Decision Response to PA Request tab.

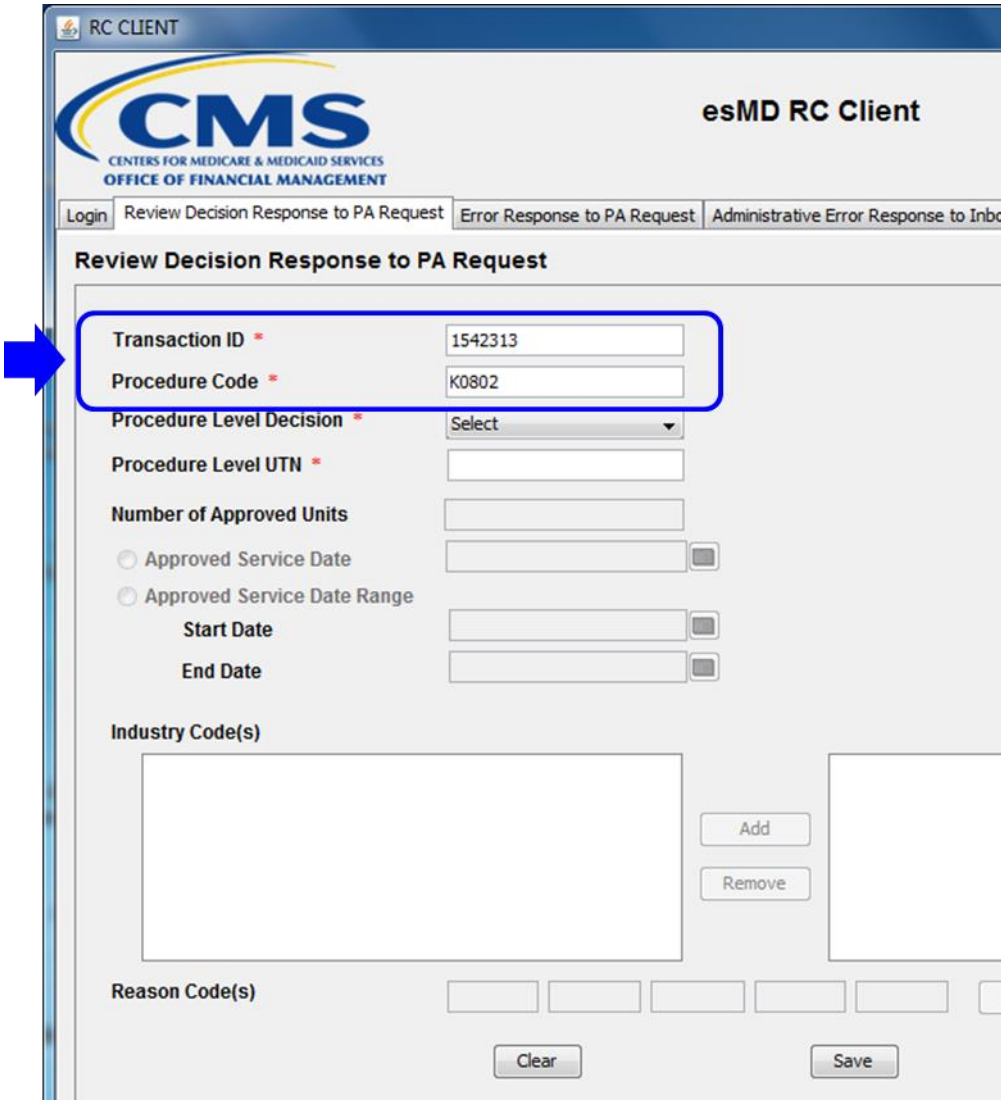
Step 2.

Entering a **N-Non Affirmed** Decision

The fields for the **Review Decision Response to PA Request** tab are displayed.

❖ Before You Begin: If you need a brief description of any of the fields on the tabs, see Appendix A: Description of Fields on RC Client Tabs on page 96.

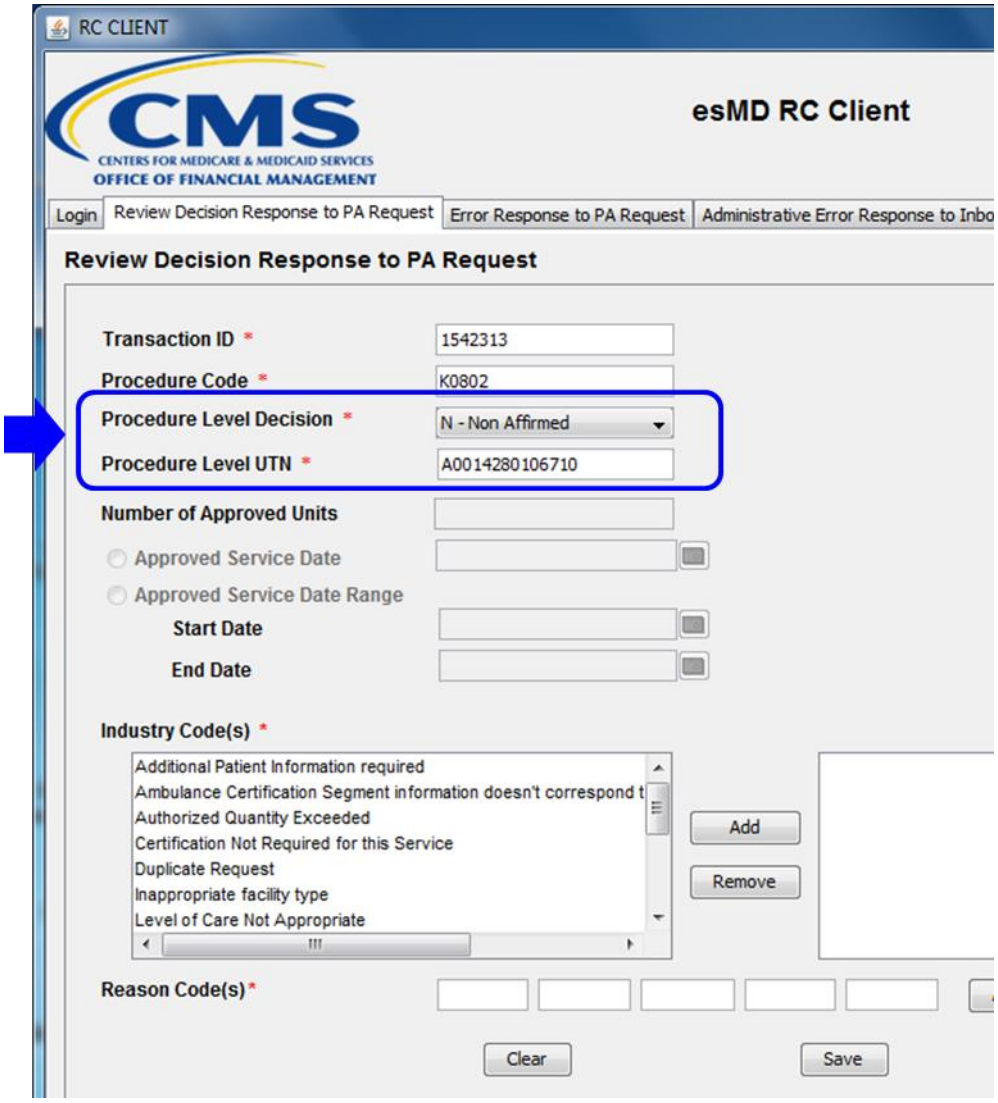
Enter the **Transaction ID** and the **Procedure Code**.



Step 3.

Entering a **N-Non Affirmed** Decision

Select the **N-Non Affirmed** decision from the **Procedure Level Decision** drop down menu and enter the **Procedure Level Universal Tracking Number (UTN)**.



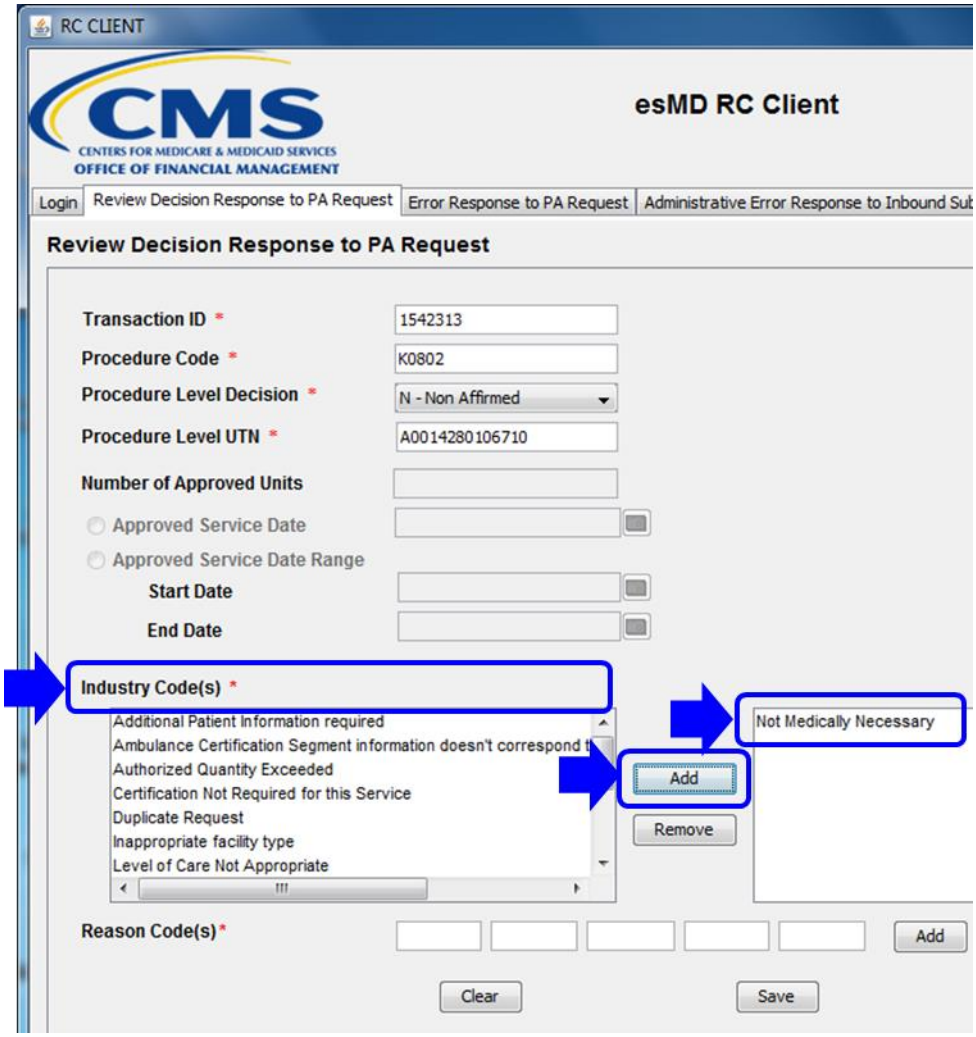
Step 4.

Entering a **N-Non Affirmed** Decision

Select an **Industry Code** from the list of Industry Codes available and then select **Add.** This will move the selection to the list of Industry Codes to be included in the response. Continue to select an **Industry Code** and then **Add**, until all of the Industry Codes have been added for this N-Non Affirmed decision.

In the example below, the "Not Medically Necessary" Industry Code will be the Industry Code added to this N-Non Affirmed decision.

Note: The Appendix C: Industry Codes provides a list all of the Industry Codes available in esMD. A maximum of five Industry Codes can be selected for a Review Decision Response to PA Request.



Step 5.

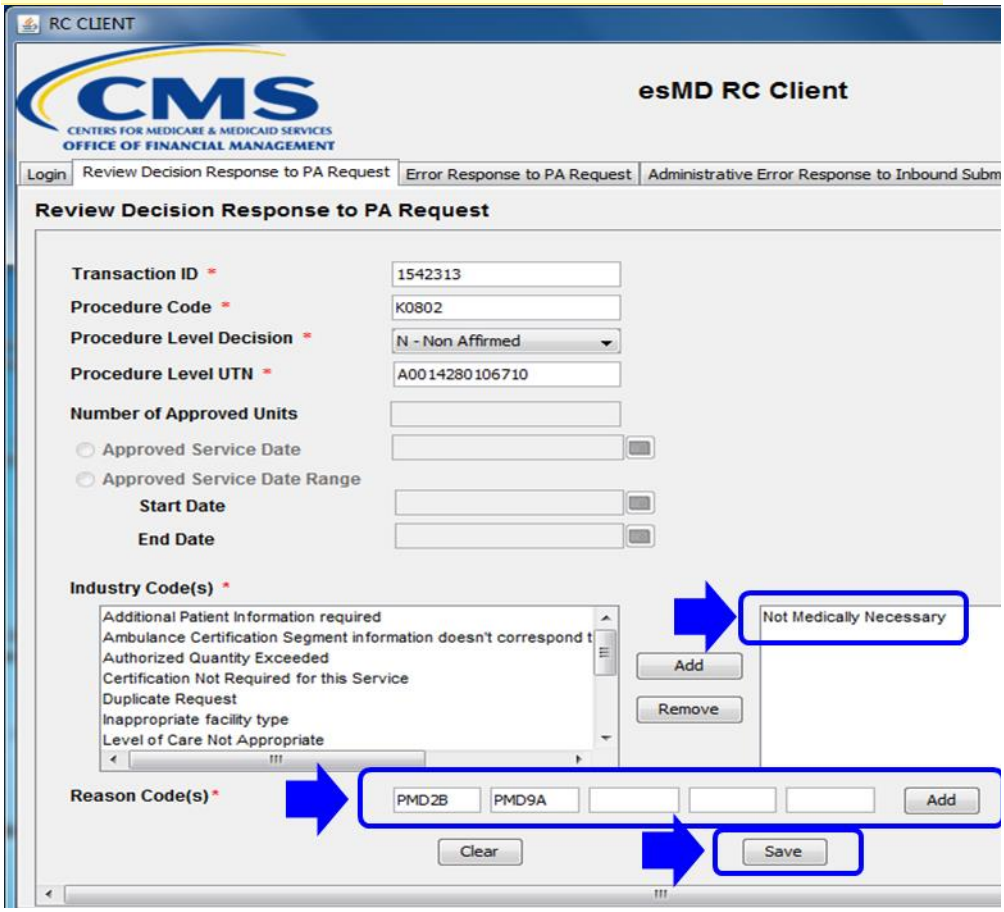
Entering a **N-Non Affirmed** Decision

After selecting Add, the Industry Code is added to the N-Non Affirmed decision, as illustrated below.

Enter the **Reason Code** or **Reason Codes** and then select **Save** to save the N-Non Affirmed decision for submission. You may enter a maximum of 25 Reason Codes.

For information on how to access an up-to-date list of PA Reason Codes, see Appendix D: PA Reason Codes.

Technical Note: After selecting Save, an Extensible Markup Language (XML) message will created to be sent to the esMD system and will be packaged into a compressed zip file. The zip file will be placed in a directory specified in the OutboundConfig/outputDirectory of the esmd-rc-client-config.xml. The outbound thread running on the RC Client will push the file to the TIBCO Managed File Transfer (MFT) server.



Step 6.

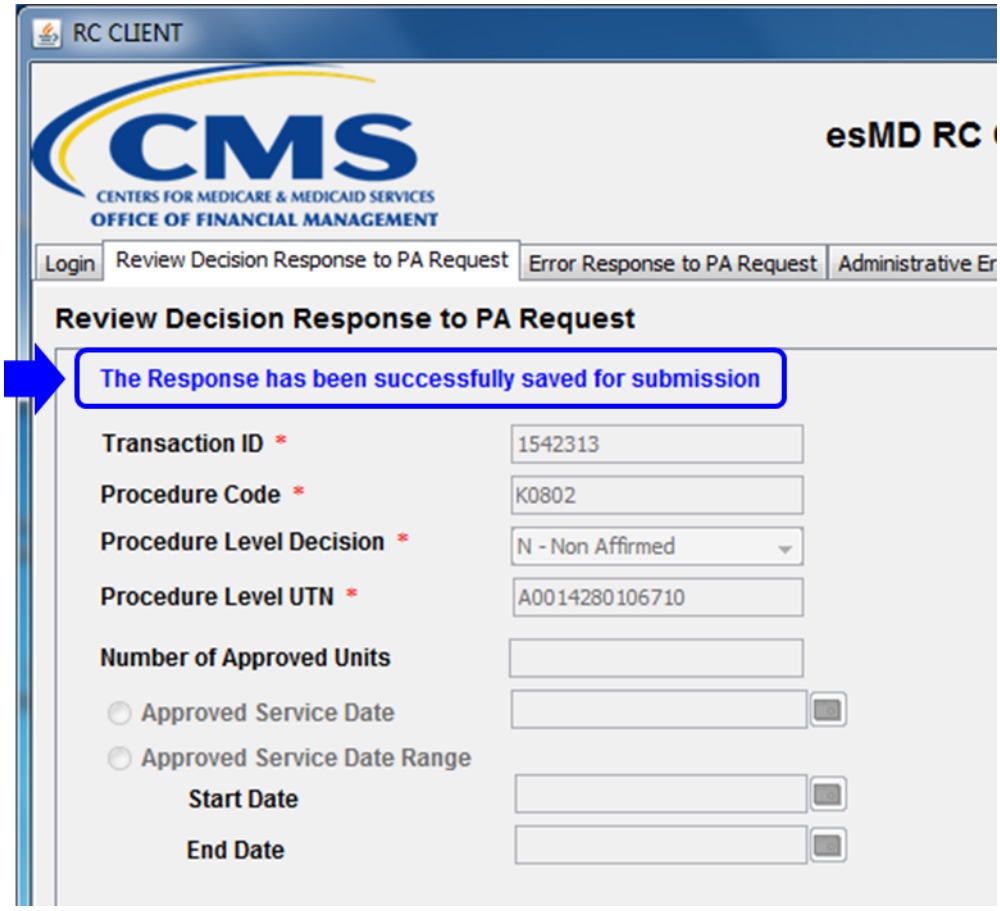
Entering a **N-Non Affirmed** Decision

After selecting Save, the "**The Response has been successfully saved for submission**" message is displayed.

Technical Note: After selecting Save, the RC Client validates the data entered and displays errors messages, as applicable.

If the data validation is successful, the N-Non Affirmed decision is created, and the "The Response has been successfully saved for submission" message is displayed.

Note: After successfully saving a decision for submission, all information in the fields are cleared and another response may be entered.



5. How to Enter an Error Code on the Error Response to PA Request Tab

This section provides step-by-step instructions on how to enter an error code on the **Error Response to PA Request** tab.

Step 1.

Entering an **Error Code**

Select the **Error Response to PA Request** tab.

❖ After a successful log in, another log in is not required to navigate to and use the Error Response to PA Request tab.

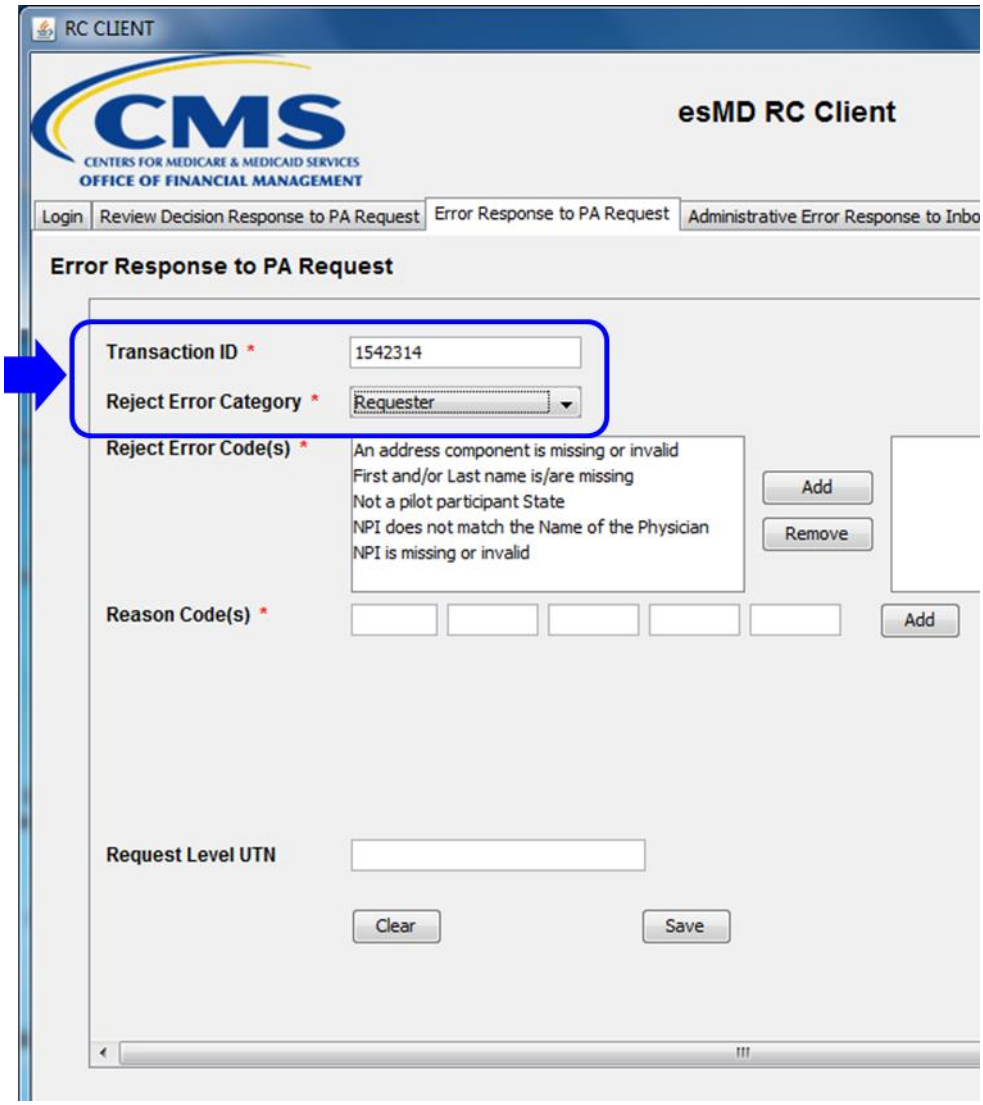
Step 2.

Entering an **Error Code**

The fields for the **Error Response to PA Request** tab are displayed.

❖ Before You Begin: If you need a brief description of any of the fields on the tabs, see Appendix A: Description of Fields on RC Client Tabs on page 96.

Enter the **Transaction ID** and select a **Reject Error Category**.

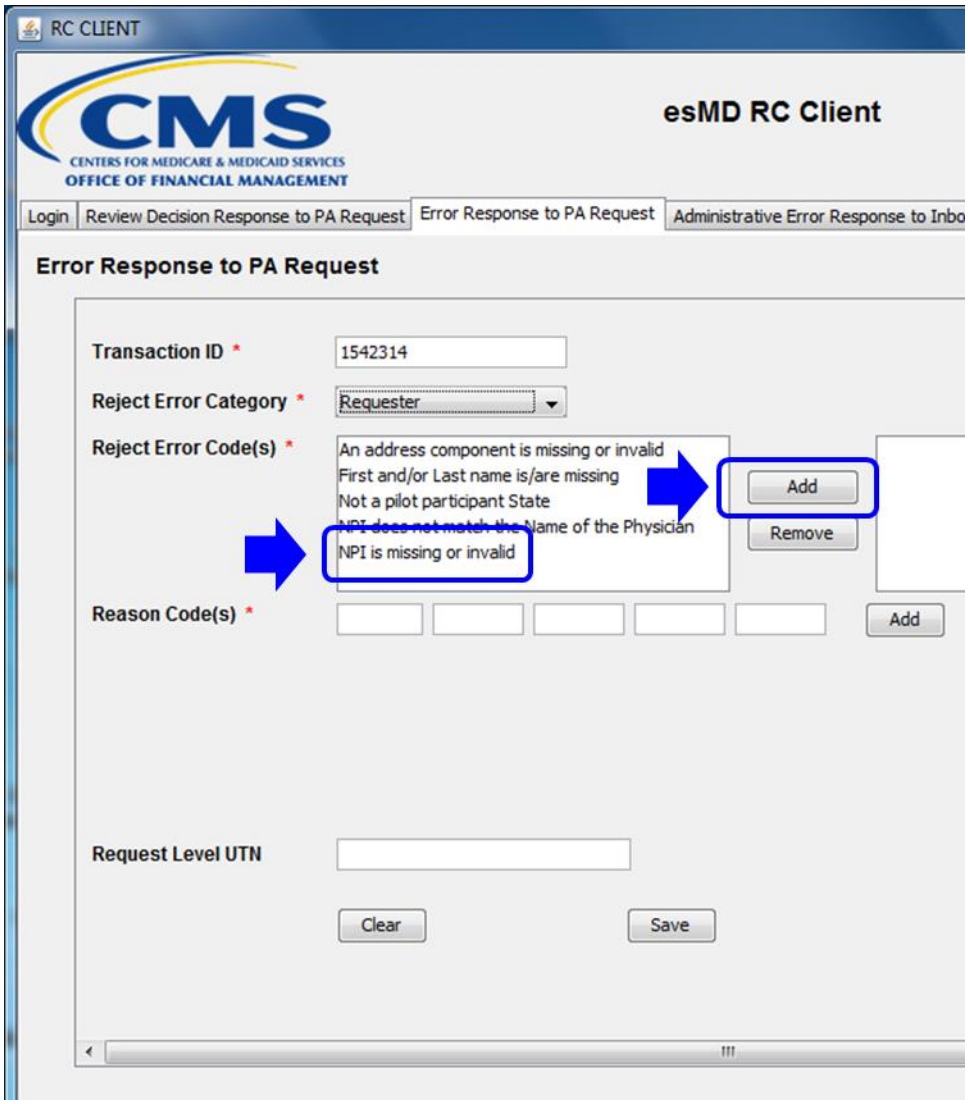


Step 3.

Entering an **Error Code**

Select a **Reject Error Code** and then **Add** to add the Reject Error Code.

For information on how to access an up-to-date list of Reject Error Codes, see Appendix B: Reject Error Codes.



Step 4.

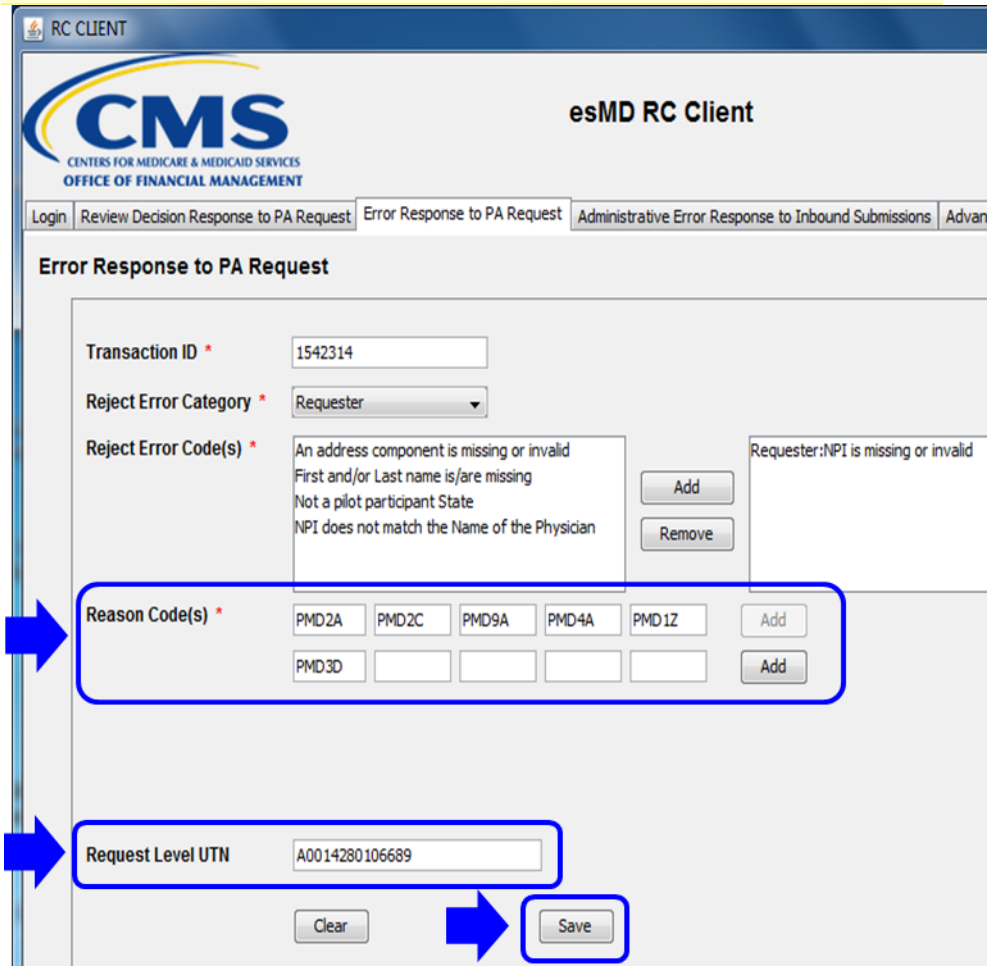
Entering an **Error Code**

Enter the **Reason Code** or **Reason Codes**. Select Add at the end of the row of Reason Code fields to add additional rows of Reason Codes, as needed.

Enter the **Request Level UTN** and then select **Save** to submit the Error Code for submission.

For information on how to access an up-to-date list of Reason Codes, see Appendix D: PA Reason Codes.

Technical Note: After selecting Save, an Extensible Markup Language (XML) message will be created to be sent to the esMD system and will be packaged into a compressed zip file. The zip file will be placed in a directory specified in the OutboundConfig/outputDirectory of the esmd-rc-client-config.xml. The outbound thread running on the RC Client will push the file to the TIBCO Managed File Transfer (MFT) server.



tep 5.

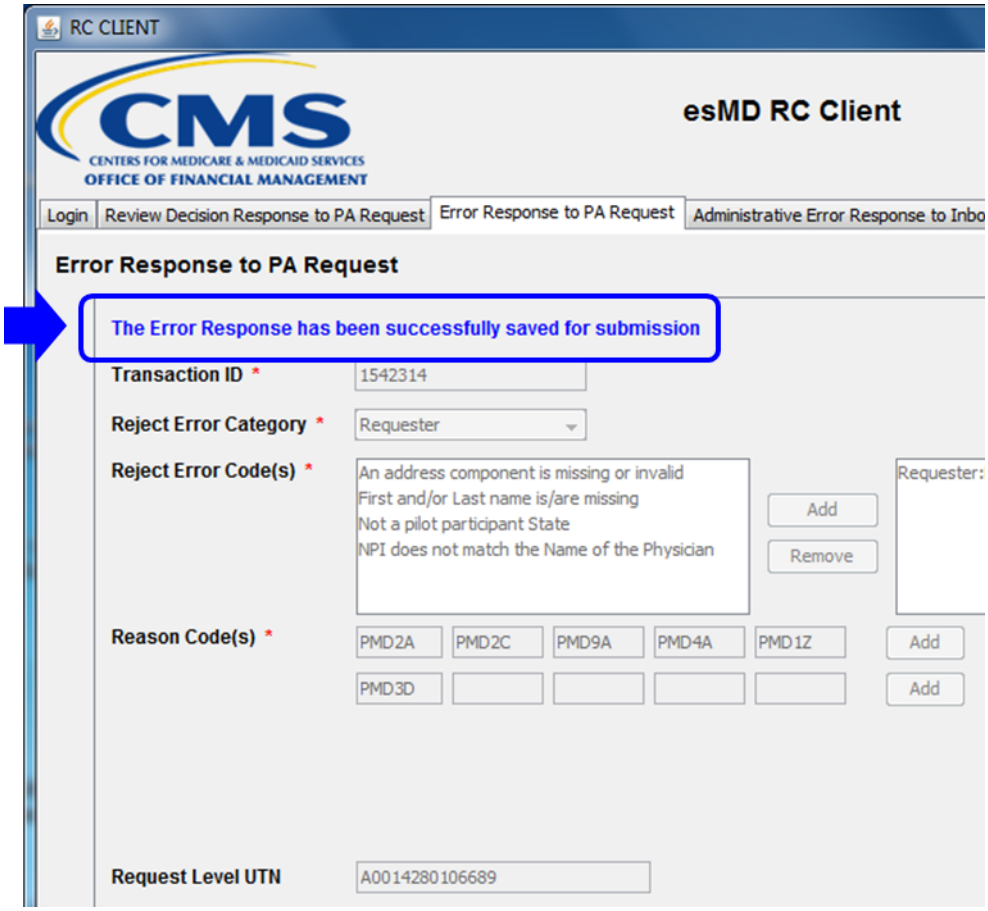
Entering an **Error Code**

After selecting Save, the "**The Error Response has been successfully saved for submission**" message is displayed.

Technical Note: After selecting Save, the RC Client validates the data entered and displays errors messages, as applicable.

If the data validation is successful, the Error Code is created, and the "The Error Response has been successfully saved for submission" message is displayed.

Note: After successfully saving a decision for submission, all information in the fields are cleared and another response may be entered.



6. How to Submit an Inbound Submission Error on the Administrative Error Response to Inbound Submissions Tab

This section provides step-by-step instructions on how to enter an inbound submission error on the **Administrative Error Response to Inbound Submissions** tab.

Step 1.

Entering an **Inbound Submissions Error**

Select the **Administrative Error Response to Inbound Submissions** tab.

❖ After a successful log in, another log in is not required to navigate to and use the Administrative Error Response to Inbound Submissions tab.

Step 2.

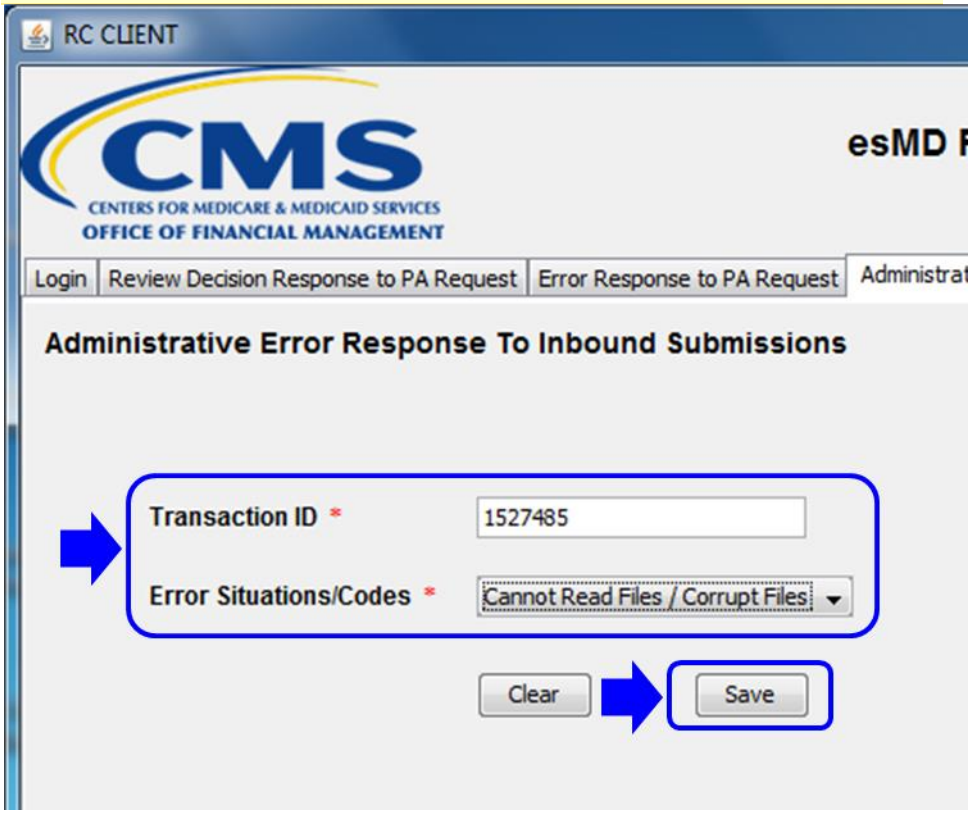
Entering an **Inbound Submissions Error**

The fields for the **Administrative Error Response to Inbound Submissions** tab are displayed.

❖ Before You Begin: If you need a brief description of any of the fields on the tabs, see Appendix A: Description of Fields on RC Client Tabs on page 96.

Enter the **Transaction ID**, select an **Error Situation** or **Error Code** from the Error Situations/Codes drop down menu, and then select **Save** to submit the Inbound Submissions error for submission.

Technical Note: After selecting Save, an Extensible Markup Language (XML) message will be created to be sent to the esMD system and will be packaged into a compressed zip file. The zip file will be placed in a directory specified in the OutboundConfig/outputDirectory of the esmd-rc-client-config.xml. The outbound thread running on the RC Client will push the file to the TIBCO Managed File Transfer (MFT) server.



Step 3.

Entering an **Inbound Submissions Error**

After selecting Save, the "**The Administrative Error Response to Inbound Submissions has been successfully saved for submission**" message is displayed.

Technical Note: After selecting Save, the RC Client validates the data entered and displays errors messages, as applicable.

If the data validation is successful, the Inbound Submissions Error is created, and the "The Administrative Error Response to Inbound Submission has been successfully saved for submission" message is displayed.

Note: After successfully saving a decision for submission, all information in the fields are cleared and another response may be entered.

7. How to Verify Connection to TIBCO MFT server, Using the Advanced/Debugging Tab

This section provides step-by-step instructions on how to verify connection to the TIBCO MFT server, using the **Advanced/Debugging** tab.

Step 1.

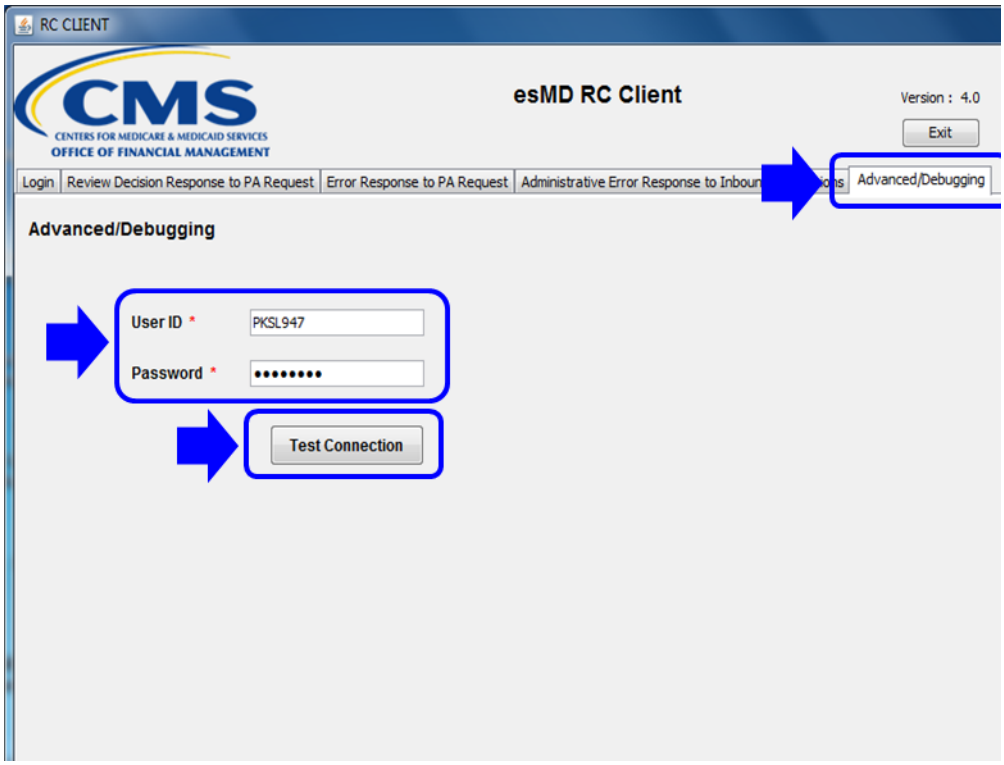
Checking Connection to TIBCO MFT Server

Select the **Advanced/Debugging** tab.

The Advanced/Debugging tab fields are displayed.

On the Advanced/Debugging tab, enter your EIDM **User ID** and **password**. (This is required on the Advanced/Debugging tab.)

Select **Test Connection**.

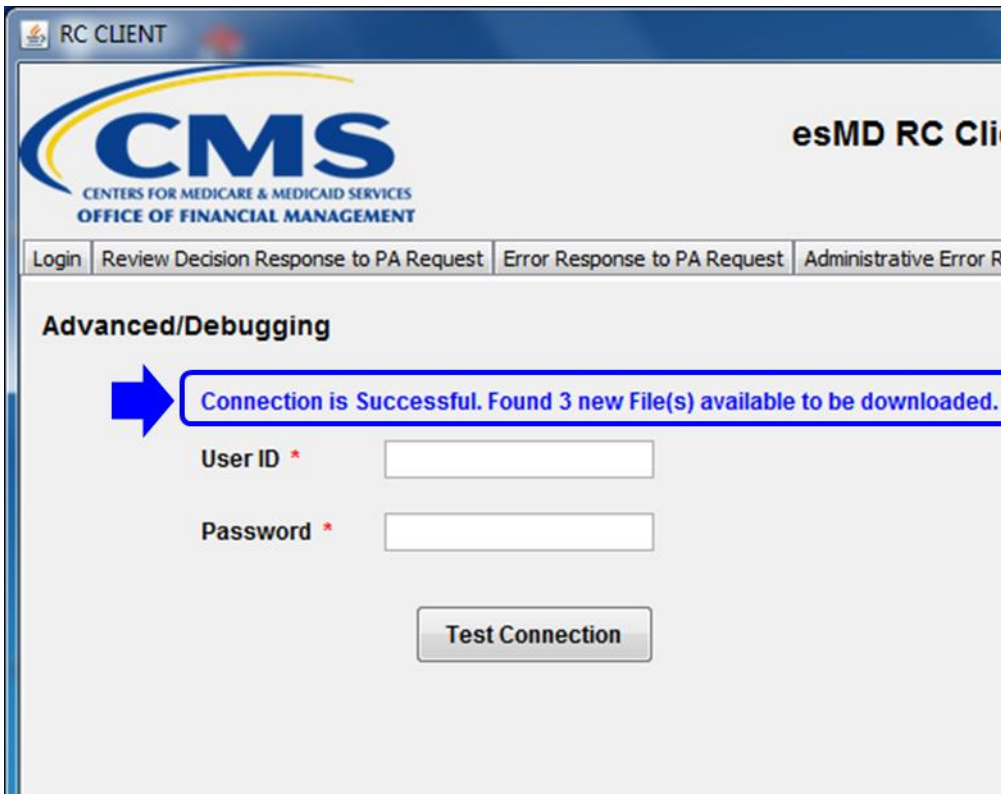


Step 2.

Checking Connection to TIBCO MFT Server

After selecting Test Connection, the "**Connection is Successful. Found 3 new File(s) available to be downloaded.**" message is displayed.

Note: After successfully testing your connection, you may select another tab.



8. System Requirements

The following are the system requirements for installing a Java version of the RC Client.

8.1 Processor

The RC Client requires a Pentium 2 266-Megahertz (MHz) processor or greater.

8.2 Disk Space

The disk requirement for the RC Java Client is 10 Megabytes (MB). The documents the RC Client pulls from the TIBCO MFT server may require additional disk space.

8.3 Memory

The RC Java Client requires a minimum of 50 MB of free memory.

8.4 Permissions

The RC Client must have read, write, and execute permissions on all the directories under the installation home.

8.5 Network

The RC Client requires internet connectivity that supports more than 32-Kilobits Per Second (Kbps) transfer speeds.

8.6 Java Framework

The RC Client requires Java Runtime Environment (JRE) 1.6 or greater to run properly.

8.7 Libraries

The Table 2: Libraries lists all the third-party libraries used by the RC Client along with their corresponding versions and a brief description of how the RC Client uses them.

Table 2: Libraries

Library	Version	Description
commons-codec	0.1.51	Used for Encoded and Decoding
commons-compress	1.2.17	Used for Extraction and compression of the packages.
commons-io	2.4	Used for reading and writing files to the Filesystem.
commons-lang	1.7	Used by the Java Secure Channel for Helper Utilities
commons-logging	1.7	Logging Framework used by the Jsch.
Jcalendar	1.0	Used for the popup Calendar in the GUI
Jsch	1.1	Java Secure Channel for SFTP Connection
log4j	4.0	Logging Framework

9. How to Install and Configure a Java Version of RC Client

Review the System Requirements in Section 8. System Requirements to make sure the machine that will host the RC Client meets the necessary requirements.

You can install the RC Client in two ways:

1. Out of the box; or
2. Custom RC Client (Java).

9.1 Out-of-the-Box

The RC Java Client API comes packaged with a sample client. To run this sample client out-of-the-box, the RCs need to follow the procedures in the following sections.

9.1.1 Keystore Set Up

Important: The RC Client uses asymmetric encryption to store the EIDM user credentials securely. For this encryption to work, you will need a secure Java KeyStore (JKS) with Public and Private keys of 2048 length. If you already have a JKS, you only need to update the configuration file with this information. Please refer to Section 15.1 Security for more details on the Security framework used by the RC Client.

1. If you do not have a JKS, create one for the RC Client to use. (Required.)
2. Type the following command to create a new keystore for the RC Client.

```
keytool -genkey -keyalg RSA –keystore <keystore> -alias <alias> -storepass <storepassword> -
keypass <keypassword> -dname "CN=<commonName>, OU=<organizationalUnit>,
O=<organizationName>, L=<localityName>, S=<stateName>, C=<country>" -keysize 2048 -
validity 360
```

Note: Replace <parameter> with the value of the parameter from the list in Table 3: Keystore Creation Parameters.

This command creates the Public and Private keys, using the Rivest, Shamir & Adleman (RSA) Algorithm with a key size of 2048 and validity of one year.

Important: After the Public and Private keys have expired, you must re-create both keys to continue to use the RC Client.

Table 3: Keystore Creation Parameters

Where	Means
<keystore>	The keystore is the home location. If you do not specify the <keystore> option, the default keystore file named *keystore.jks* in the user's home directory will be created, if it does not already exist. For example, the config/keystore.jks will be created.
<alias>	The certificate chain and the private key are stored in a new keystore entry, identified by *alias.*
<storepassword>	The store password is used to protect the integrity of the keystore. It must be at least six characters long.
<keypassword>	The key password is used to protect the private key of the generated key pair. If a password is not provided by the user, the user is prompted to provide it. If you press Enter at the prompt for the key password, the key password is set to the same password that was used for the keystore. The <keypassword> must be at least six characters long.
<commonName>	The common name is the name for any entity, such as the name of a person (for example, Susan Jones) or the name of your company.
<organizationalUnit>	The organizational unit can be used for a small organization, department, or division of an organization (for example, Purchasing).
<organizationName>	The organization name is for a large organization or company (for example, ABC Systems, Inc.).
<localityName>	The locality name can be for a city (for example, Palo Alto).
<stateName>	The state name can be for a U.S. state or province of another country (for example, California or Ontario in Canada).
<country>	The country is a two-letter code (for example, US).

9.1.2 Integrity Verification

The command below will print the public key from the keystore and verify the keystore integrity.

1. Type the following command:

```
keytool -list -v -keystore <keystore> -storepass <storepassword>  -alias <alias>
```

Note: Replace <parameter> with the value for the parameter listed in Table 3: Keystore Creation Parameters.

9.1.3 Java Cryptography Extension (JCE) Policy Update

In addition to creating and providing the keystore, you may need to override the Java Cryptography Encryption (JCE) security policy files if these files were not already overridden.

9.1.3.1 Understanding the JCE Security Policy Files

Due to import control restrictions, the version of the JCE security policy files that are bundled in the Java Development Kit™ (JDK) environment allow "strong" but limited cryptography to be used. To run the RC Client, this security policy must be overridden with the "unlimited strength"

policy files that contain no restrictions on cryptographic strengths. If the RC Client is run with the default JCE security policy files, it will cause an error similar to the following:

```
java.security.InvalidKeyException: Illegal key size at
javax.crypto.Cipher.a(DashoA13*...)
```

New JCE security policy files are packaged along with the RC Client and are in the “setup” subdirectory of the installation directory.

Note: These files do not contain additional encryption functionality because such functionality is supported in Sun's JDK.

9.1.3.2 Understanding the Export/Import Issues

JCE for JDK has been through the U.S. export review process. The JCE framework, along with the Sun JCE provider that comes standard with it, is exportable. The JCE architecture allows flexible cryptographic strength to be configured via jurisdiction policy files. Due to the import restrictions of some countries, the jurisdiction policy files distributed with the JDK software have built-in restrictions on available cryptographic strength.

9.1.3.3 JCE Policy Files

The setup directory in the RC Client installation contains the policy files listed in Table 4: JCE Policy Files.

Table 4: JCE Policy Files

Policy File	Description
local_policy.jar	Unlimited strength local policy file
US_export_policy.jar	Unlimited strength U.S. export policy file

9.1.3.4 Installation Locations for Windows and UNIX

<java-home> refers to the directory where the JRE was installed. It is determined based on whether you are running JCE on a JRE with the JDK installed. The JDK contains the JRE, but at a different level in the file hierarchy. Table 5: Java Development Kit and Table 6: Java Runtime Environment show examples of the installation for Java version 1.6, but this will work for Java version 7, as well.

Table 5: Java Development Kit

Environment	Example JDK Installation Directory	JAVA_HOME
Windows	C:\jdk1.6.0	C:\jdk1.6.0\jre
Unix	/home/user1/jdk1.6.0	/home/user1/jdk1.6.0/jre

Table 6: Java Runtime Environment

Environment	Example JRE Installation Directory	JAVA_HOME
Windows	C:\jre1.6.0	C:\jre1.6.0
Unix	/home/user1/jre1.6.0	/home/user1/jre1.6.0

Notes:

1. UNIX (Solaris/Linux) and Windows use different pathname separators; use the appropriate one ("\" or "/") for your environment; and

2. On Windows, for each JDK installation, there may be an additional JRE installed under the "Program Files" directory. Ensure you install the unlimited strength policy Java Archive (JAR) files for all JREs that you plan to use.

9.1.3.5 Setting Up Encryption/Decryption without Limitation

To use the encryption/decryption functionalities of the JCE framework without any limitation:

1. Make a copy of the original JCE policy files (US_export_policy.jar and local_policy.jar in the standard place for JCE jurisdiction policy JAR files) in case you later decide to revert to these “strong” versions; and

2. Copy the policy files with the unlimited strength versions from the “setup” directory per the version of Java to be used (java6 or java7) under the installation directory to the security directory shown in Table 7: Security Directory.

Table 7: Security Directory

Environment	Installation Directory
Windows	<java-home>/lib/security
Unix	<java-home>\lib\security

9.1.4 Configuring the RC Client

Once the keystore is created and the policy files are installed, the RC Client is ready to be configured to use the keystore.

1. Update the keystore information in the configuration file (Required); and

   Important: The XML configuration file (i.e. config/esmd-rc-client-config.xml) is used by the RC Client to retrieve important configuration parameters necessary for its operation.

2. Use the comments for each configuration parameter shown in Table 8: Sample RC Client Configuration File as a guide in entering your data.

Table 8: Sample RC Client Configuration File

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:ESMDConfig xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config"
xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config esmd-config.xsd
">

    <!--The TIBCO MFT server Configuration-->
    <ESMDSFTPServer>
        <!--TIBCO MFT Sever host name or IP -->
        <host>eftp2.cms.hhs.gov</host>
        <!--The TIBCO MFT SFTP PORT-->
        <port>11022</port>
        <!-- Update: Use T for VAL, P for PROD-->
        <environmentId>T</environmentId>
        <!--The EFT File Name Prefix-->
        <eftFilePrefix>#EFT</eftFilePrefix>
    </ESMDSFTPServer>
    <!--The Keystore information for Encryption and Security-->
    <KeyStoreInfo>
        <!-- Update: The JKS Keystore Path-->
        <keyStoreLocation>/RCClient/config/keystore.jks</keyStoreLocation>
        <!-- Update: The Encrypted Keystore Password-->
        <encKeyInfo>ItwdafsdviaZNpvV54aRM9ZzQiw==</encKeyInfo>
        <!-- Update: The Encrypted Private Key Password-->
        <encKeyInfoExt>srs8adsfasRtLEB2I=</encKeyInfoExt>
        <!-- Update: The Certificate Alias-->
        <certAlias>selfsigned</certAlias>
    </KeyStoreInfo>
    <!--The Inbound Process Configuration-->
    <InboundConfig>
        <!-- Update: Enable the Inbound Process? true/false-->
        <enabled>true</enabled>
        <!--The Pull Frequency for the Inbound Process in minutes; the
default is 240 minutes i.e., 4 hours-->
        <checkFrequency>30</checkFrequency>
        <!-- Update: The RC Client installation/home directory-->
        <rcHomeDirectory>/RCClient</rcHomeDirectory>
        <!-- Update: The target directory to extract the downloaded inbound
files before routing-->
        <targetDirectory>/RCClient/data/download</targetDirectory>
```

```
        <!-- Update: The input directory where the inbound payloads and the
metadata will be routed after the extraction-->
        <inputDirectory>/RCClient/data/input</inputDirectory>
        <!-- Update: The temp directory where the files are pulled from
TIBCO MFT-->
        <tempDirectory>/RCClient/data/temp</tempDirectory>
        <!-- Update: The Error directory for routing the inbound error
notifications from esMD/HIH-->
        <errorDirectory>/RCClient/data/error</errorDirectory>
        <!-- Update: The configuration directory for RC Client-->
        <configDirectory>/RCClient/data/conf</configDirectory>
        <!-- Update: The notifications directory for routing the inbound
notifications from esMD/HIH-->

<notificationsDirectory>/RCClient/data/notification</notificationsDirectory>
        <!-- Update: The Remote Inbound Directory path on the TIBCO MFT
server-->
        <remoteInboundDir>/ES####</remoteInboundDir>
        <!--Update: The mail box number for the inbound files used to pick
the inbound files to pull-->
        <inboundRoutingId>ES####</inboundRoutingId>
    </InboundConfig>
    <!--The Outbound Process Configuration-->
    <OutboundConfig>
        <!-- Update: Enable the Outbound Process? true/false-->
        <enabled>true</enabled>
        <!--The push frequency for the Outbound process in minutes default
is 15 minutes-->
        <pushFrequency>15</pushFrequency>
        <!-- Update: The temp directory to use for the outbound process for
creating the PMPDA/Notification files-->
        <tempDirectory>/RCClient/data/temp</tempDirectory>
        <!-- Update: The local outbound directory to push the outbound files
from-->
        <outputDirectory>/RCClient/data/output</outputDirectory>
        <!-- Update: The Remote Outbound directory to push files-->
        <remoteOutboundDir>/ES####_UPLOAD</remoteOutboundDir>
        <!--The Remote Outbound mail box number to push files onto esMD
servers via TIBCO MFT-->
        <outboundRoutingId>ESMD2</outboundRoutingId>
        <!--The Outbound File name prefix-->
        <outboundFilePrefix>ON</outboundFilePrefix>
    </OutboundConfig>
</ns2:ESMDConfig>
```

9.1.4.1 Configuring Your Password Encryption

1. Run the encryptConfig.bat script to update the KeystoreInfo section with the encrypted keystore and private key password;
2. When the script prompts, enter your keystore and private key passwords, as shown in Figure 2: Keystore Password Encryption and Figure 3: Private Key Password Encryption, and click **OK** in each Input window; and

Figure 2: Keystore Password Encryption

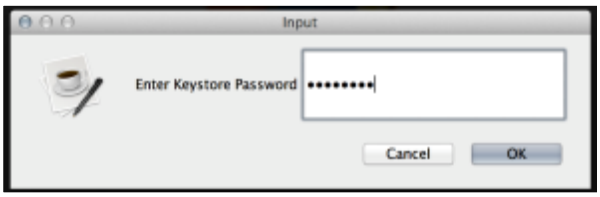


Figure 3: Private Key Password Encryption

3. Update the XML configuration file parameter "`certAlias`" with the alias of the certificate you created in Section 9.1.1 Keystore Set Up .

   The KeystoreInfo section of the XML Configuration file is now updated with the encrypted passwords and the certificate information required for the RC Client operation.

9.1.5 Running the RC Client

Before you, as the RC, run the sample RC Client, you must double-check all the configuration parameters in the XML configuration file, especially the ones with the "Update" prefix in the comments of the sample XML configuration file, as shown in Table 8: Sample RC Client Configuration File.

1. To run the sample RC Client, run the "RcClientUI.exe" utility provided in the distribution package.
2. Start the RC Client by providing login credentials for the Login tab and select the “Login and Run RC Client” button.

9.2 Custom RC Client

The RC Java Client provides an API, so the RC can extend the RC Client to fit the RC’s environmental needs. The API enables the RC to perform the following functions:

- Log in to the TIBCO MFT server (See Section 1.4.3 RC Client Operation Overview);
- Get Notifications from the TIBCO MFT server using the Secure Shell (SSH) File Transfer Protocol (SFTP). (Refer to Section 15.2.2 Inbound);
- Decrypt/encrypt and store the login credentials using a secure RSA algorithm. (Refer to Section 15.2.7 Utilities - Encryption);

- Pull medical documentation from the TIBCO MFT server. (Refer to Section 15.2.2 Inbound);
- Extract the downloaded packages. (Refer to Section 15.2.2 Inbound);
- Check the payloads using checksums in the metadata. (Refer to Section 15.2.2 Inbound);
- Push the outbound files from the “output” directory. (Refer to Section 15.2.3 Outbound); and
- Create custom files (for example, the custom PMD PA files. Refer to Section 15.2.4).

Note: The procedures for customizing the RC Client API are beyond the scope of this document. (The source code that will be packaged along with the RC Client contains the documentation needed for integrating the API.)

10. TIBCO® MFT File Transfers

The RC Client uses a TIBCO MFT server to interact with esMD. It uses the SFTP to connect to the TIBCO MFT server and uses the ls/Get/Put commands to interact with the files. There are four types of inbound files that RC Client pulls from the TIBCO MFT server, described in Table 9: Inbound Files.

Note: "ES0001" is a sample mailbox number that the TIBCO MFT server uses to identify the RC and "0977890" is a sample Transaction ID, which is shown in Table 9: Inbound Files and Table 10: Outbound Files. The final two qualifiers in the file name that are prefixed with "D" and "T" are the Date and Timestamp, respectively. The Validation files will have a "T" prefix and the Production files will have a "P" prefix.

Only 1,022 files will be visible in the TIBCO MFT server at one time by the MFT Mailbox Routing number. As each file is pulled, the TIBCO MFT server will bring new files from the mainframe and place them at the bottom of the queue.

Table 9: Inbound Files

Type	Example File Name	Delivery Type Description
Inbound	T.ES0001.**E**0977890.D140116.T1033445	The E in prefix to the 0977890 Transaction ID indicates an esMD payload.
Inbound	T.ES0001.**A**0977890.D140116.T1033445	Successful Acknowledgement that the HIH has received the pickup notification.
Inbound	T.ES0001.**R**0977890.D140116.T1033445	Validation Error received by the RC for the Pickup Notification sent to esMD.
Inbound	T.ES0001.**N**0977890.D140116.T1033445	Successful Acknowledgement that the HIH has received the PA Review Response.
Inbound	T.ES0001.**M**0977890.D140116.T1033445	Validation Error received by the RC for the Administrative Error Response sent to esMD.
Inbound	T.ES0001.**V**0977890.D140116.T1033445	Validation Error received by the RC for the PA Review Response sent to esMD.
Inbound	T.ES0001.**X**0977890.D140116.T1033445	Virus Scan Failure Error received by the RC from the esMD on a file sent previously.
Inbound	T.ES0001.**Y**0977890.D140116.T1033445	McAfee Gateway down Error for outbound processing received by the RC from the esMD on a file sent previously.
Inbound	T.ES0001.**S**0977890.D140116.T1033445	Successful Acknowledgement that the HIH has received the administrative error response.

Table 10: Outbound Files

Type	Example File Name	Delivery Type Description
Outbound	T#EFT.ON.ESMD2.**E**0977890.D140116.T1033445	E indicates PA Review Response.
Outbound	T#EFT.ON.ESMD2.**D**0977890.D140116.T1033445	D indicates an Administrative Error.
Outbound	T#EFT.ON.ESMD2.**P**0977890.D140116.T1033445	P indicates an esMD pickup notification i.e. both and success and error.

11. XML Messages

This section describes the various XML messages transferred during the inbound and outbound processes.

11.1 Inbound

Note: Please refer to Section 13.3 Inbound Processes for details on how RC Client routes the inbound files once they are successfully processed into the data directories.

The RC Client transfers the following files during the inbound process:

- Payload Files;
- Flat File Rendering (FFR);
- Cover Sheet;
- Metadata File;
- Pickup HIH Status Response;
- Pickup Validation Error Response;
- Administrative Error HIH Status Response;
- Administrative Error Response Validation Error;
- Virus Scan Error Response;
- PA Review Result HIH Status Response; and
- PA Review Result Validation Error Response.

11.1.1 Payload Files

The RC Client will receive PDF files as payloads in the inbound documents with delivery type "E". An example payload file name is E_185457-esmdQSSIVG0407141396893280928-0.pdf.

11.1.2 Flat File Rendering (FFR)

The RC Client will receive FFR file in the inbound documents along with the payload files for the X12N 278 request. An example FFR file is shown in Table 11: E_185457-flatfilerendering.ffr.

For more details on the layout of the FFR and X12N 278 requests, please see the X12N 278 Companion Guide under the esMD Downloads section, using the link below:

http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Information_for_Review-Contractors.html

Table 11: E_185457-flatfilerendering.ffr

```
ST                2781523     005010X217                          BHT
0007013920394930203                                   200812081615    RUHL
1           201NM1               X3 2JOHNSON
46111222333
HL                2           1           211NM1               1P 1SMITH
MUFFY                              M                                  PH.D
XX1111111112
N3                PO BOX 123
N4                CINCINNATI                    OH43017
PER               ICWILBER
FX8189991234
TE8187691304
EXPRV               CO PXC208D00000X
HL                3           2           221NM1               IL 1JONES
BARBARA                            T                       MR        M.D.
IIAA123456
N3                PO BOX                                                 123
N4                CINCINNATI                    OH43017
DMG               D8 19511204                           FHL                4
3           EV1TRN               1 12345678900987654321768958473
1311234567500                                            UM
INI1 13A AA EM AP ONCA U  34Y1 PWK               77ELACATTACHMENT CONTROL
NUMBER 1
NM1               72 1JOHNSON
BARBARA
46111111111
N3                PO BOX 123
N4                CINCINNATI                    OH43017                HL
5           4           SS0TRN               1 0001-201501150001UBERTEST-
20141224-SVC1           9555555555UBERTESTHAPPYPATH
UM                SCN3     SV1               HC27447
36345.25          UN1                        SE                27
1523
```

11.1.3 Cover Sheet

The RC Client will receive cover sheet in the inbound documents along with the payload files for the X12N 278 request. An example cover sheet is shown in Figure 4: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 1 , Figure 5: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 2, and Figure 6: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 3.

Figure 4: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 1

EDI Cover Sheet

esMD Transaction ID :	1200809
Procedure code Qualifier :	HCPCS
Procedure code :	A0428
Pricing Modifiers :	
Pricing Modifier 1 :	25
Pricing Modifier 2 :	21
Pricing Modifier 3 :	23
Pricing Modifier 4 :	55
Beneficiary identification :	
HICN :	215123556A
Date of Birth (ccyymmdd):	19511204
Gender :	M
Last Name :	AMBLSUBSCRIBERLASTNAME
First Name :	AMBLSUBSCRIBERFIRSTNAME
Address Line 1 :	AMBBENEFICIARY LOOP2010C ADDR LINE 1
Address Line 2 :	SUB ADDR2
City :	WINDSORMILL
State :	MD
Zip :	21244
Ordering Provider Identification :	
NPI :	1234567893
Last Name :	AMBORDERINGPHYLASTNAME
First Name :	AMBORDERINGPHYFNAME
Address Line 1 :	AMBSTREET ORDERING PHYSICIAN 2010EA
Address Line 2 :	
City :	WINDSORMILL
State :	MD
Zip :	21244
Rendering Provider :	
NPI :	
Last Name :	
First Name :	
Address Line 1 :	
Address Line 2 :	
City :	
State :	
Zip :	
Date Request Submitted (yymmdd):	150724
Service State :	

Date: 2016-04-14T05:05:49PM EDT *Page: 1*

Figure 5: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 2

Facility/Agency :	
NPI :	1234567893
Last Name :	FACILITY ORG NAME
Address Line 1 :	AMBSTREET SERVICE PROVIDER 2010EA
Address Line 2 :	
City :	WINDSORMILL
State :	MD
Zip :	21244
Place of Service / Type of bill code :	13
Diagnosis codes :	
Diagnosis Code Qualifier (1) :	BK
Diagnosis Code (1) :	78609
Diagnosis Code Qualifier (2) :	BF
Diagnosis Code (2) :	85135
Diagnosis Code Qualifier (3) :	BF
Diagnosis Code (3) :	8488
Diagnosis Code Qualifier (4) :	BF
Diagnosis Code (4) :	8471
Diagnosis Code Qualifier (5) :	BF
Diagnosis Code (5) :	8472
Diagnosis Code Qualifier (6) :	BF
Diagnosis Code (6) :	84510
Diagnosis Code Qualifier (7) :	BF
Diagnosis Code (7) :	8461
Diagnosis Code Qualifier (8) :	BF
Diagnosis Code (8) :	8502
Diagnosis Code Qualifier (9) :	BF
Diagnosis Code (9) :	8500
Diagnosis Code Qualifier (10) :	BF
Diagnosis Code (10) :	85106
Diagnosis Code Qualifier (11) :	BF
Diagnosis Code (11) :	8489
Diagnosis Code Qualifier (12) :	BF
Diagnosis Code (12) :	8431
Requester Information :	
NPI :	1111111112
Last Name :	REQUESTERLASTNAME
First Name :	REQUESTERFIRSTNAME
Address Line 1 :	REQUESTER STREET LOOP 2010B
Address Line 2 :	
City :	WINDSORMILL

Date: 2016-04-14T05:05:49PM EDT *Page: 2*

Figure 6: Sample EDI Cover Sheet - E_1234567-coversheet.pdf Page 3

State :	MD
Zip :	21244
Requester/Contact :	
Contact Name :	DR.AMBULANCECONTACT
Contact Telephone # :	4035556789
Number of units requested :	
Units requested :	80
Service Period :	
Service Period Requested (ccyymmdd):	20150930-20151030
Attachment Control Number :	KUMARPICKUP007

11.1.4 Metadata File

The metadata file accompanies the payload files, FFR, and X12N 278 request copy in the inbound documents with delivery type "E". The metadata file contains information about the payloads like the Object Identifier (OID), Transaction ID, Submission metadata (includes Attachment Control Number and other information), and optional metadata. The Content Type Code will change for each line of business. See Table 12: E_123456-Metadata.xml.

Note: The metadata file will remain the same for all lines of business including Additional Documentation Requests (ADRs), PMD PA Requests, Non-Emergent Ambulance Transport and HBO Prior Authorization Requests, First Level Appeal Requests, Second Level Appeal Requests, Recovery Auditor (RA) Requests, DMEPOS Requests, and HHS Requests.

Note: The Claim ID is optional for First Level Appeal Requests, Second Level Appeal Requests, and ADMCs. For more information on the Content Type Codes, please refer to Appendix G: Content Type Codes.
Note: The Case ID is required for Second Level Appeal Requests. Please refer to Appendix G: Content Type Codes.

Table 12: E_123456-Metadata.xml

```
<ns0:RetrieveMedicalDocumentationResponse
xmlns:ns0="http://esmd.ois.cms.hhs.gov/v1/rc" returnCode="1"
serviceSuccessful="true">
    <statusDescription>The RetrieveMedicalDocumentationRequest processed
successfully.</statusDescription>
    <NumberOfDocuments>3</NumberOfDocuments>
    <ESMDPackage>
        <ESMDTransaction TransactionId="123456" DeliveryType="E"/>
        <SendingOID>urn:oid:123.456.657.132</SendingOID>
```

```
        <TargetOID>urn:oid:2.16.840.1.113883.13.34.110.1.999.1</TargetOID>
        <CompleteSubmission>true</CompleteSubmission>
        <RequestType>X12-XDR</RequestType>
        <SubmissionMetadata>
            <BusinessType>XDR X12</BusinessType>
            <CreationTime>2015-08-04T17:16:16.887-04:00</CreationTime>
            <SubmissionTime>2015-08-04T17:16:16.887-04:00</SubmissionTime>
            <EFTSubmissionTime>2015-08-04T17:16:16.887-
04:00</EFTSubmissionTime>
            <ContentTypeCode>13</ContentTypeCode>
            <NPI>1111111112</NPI>
        </SubmissionMetadata>
        <Documentation DocumentUniqueIdentifier="E_1191152-
esmdQSSISKADRVALID1438722963924143872297466 1_0" MimeType="application/pdf"
FileName="E_1191152-esmdQSSISKADRVALID1438722963924143872297466 1_0.pdf">
            <OptionalMetadata>
                <FieldName>FileName</FieldName>
                <FieldValue>E_1191152-
esmdQSSISKADRVALID1438722963924143872297466 1_0.pdf</FieldValue>
            </OptionalMetadata>
            <OptionalMetadata>
                <FieldName>Description</FieldName>
                <FieldValue>From esMD</FieldValue>
            </OptionalMetadata>
            <OptionalMetadata>
                <FieldName>CheckSum</FieldName>

<FieldValue>73d1ba48402985bac6ddab12f47c179ddddbe4c6</FieldValue>
            </OptionalMetadata>
        </Documentation>
        <Documentation DocumentUniqueIdentifier="E_1191152-
flatfilerendering" MimeType="text/xml" FileName="E_1191152-
flatfilerendering.ffr">
            <OptionalMetadata>
                <FieldName>AttachmentControlNumber</FieldName>
                <FieldValue>SRNKS005</FieldValue>
            </OptionalMetadata>
            <OptionalMetadata>
                <FieldName>CheckSum</FieldName>

<FieldValue>30fecc47759d3220c52c6cd8cc14315907fe212a</FieldValue>
            </OptionalMetadata>
        </Documentation>
        <Documentation DocumentUniqueIdentifier="E_1191152-coversheet"
MimeType="application/pdf" FileName="E_1191152-coversheet.pdf">
            <OptionalMetadata>
                <FieldName>AttachmentControlNumber</FieldName>
                <FieldValue>SRNKS005</FieldValue>
            </OptionalMetadata>
            <OptionalMetadata>
                <FieldName>CheckSum</FieldName>

<FieldValue>54bb0dee19b1965059eb00a028465419bdb1dc7b</FieldValue>
            </OptionalMetadata>
        </Documentation>
    </ESMDPackage>
```

```
</ns0:RetrieveMedicalDocumentationResponse>
```

11.1.5 Pickup HIH Status Response

When the RC Client sends a pickup notification to esMD, the esMD application processes the notification and sends the response to the HIH. Once the esMD application receives the acknowledgement for the pickup notification from HIH, then it generates the Pickup Status Response and sends it to the RC, indicating the response was sent to the HIH, as detailed in the code in Table 13: A_123456_Pickup_HIH_Status_Response.xml.

Note: The HIH Pickup Status Response will remain the same for all lines of business including ADRs, PMD PA Requests, Non-Emergent Ambulance Transport, and HBO PA requests, ADMCs, DMEPOS Requests, and HHS PA Requests.

Table 13: A_123456_Pickup_HIH_Status_Response.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>

<ns2:RCPickupNotificationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config">
    <ESMDTransactionId>123456</ESMDTransactionId>
    <ErrorInfo xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:nil="true"/>
    <status>Success</status>
    <statusDesc>SENT PICKUP STATUS TO HIH</statusDesc>
</ns2:RCPickupNotificationResponse>
```

11.1.6 Pickup Validation Error Response

When the RC Client sends a Pickup Notification to esMD, the esMD application processes and sends the Pickup Notification to the HIH. If there is an error in processing the Pickup Notification submitted by the RC, the esMD application generates the Pickup Validation Error Response, as detailed in Table 14: R_123456_ Pickup_Validation_Error.xml, and sends it to the RC. The RC will correct the pickup notification and resubmits the PA Results Result. Refer to the code located in Table 14: R_123456_ Pickup_Validation_Error.xml.

Table 14: R_123456_ Pickup_Validation_Error.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<ns0:RCPickupNotificationResponse
xmlns:ns0="http://esmd.ois.cms.hhs.gov/v1/rc/config">
    <ESMDTransactionId>123456</ESMDTransactionId>
    <ErrorInfo>
        <ErrorCode>614</ErrorCode>
```

```
        <ErrorName/>
        <ErrorDescription>Unable to parse RC
response</ErrorDescription>
    </ErrorInfo>
    <Status>FAILED</Status>
    <StatusDesc>ESMD internal system error, please resubmit
proper file</StatusDesc>
</ns0:RCPickupNotificationResponse>
```

11.1.7 Administrative Error HIH Status Response

When the RC Client sends an administrative error for an inbound submission to esMD, the esMD application processes the administrative error and sends the response to the HIH. Once the esMD application receives the acknowledgement for the administrative error from HIH, then it generates the Administrative Error HIH Status Response and sends it to the RC, indicating the error was sent to the HIH, as detailed in the code in Table 15: S_123456_Administrative_Error_HIH_Status_Response.xml.

Note: The Administrative Error HIH Status Response will remain the same for all lines of business including ADRs, PMD PA Requests, Non-Emergent Ambulance Transport, and HBO PA requests, DMEPOS Requests, HHS PA Requests, and ADMCs.

Table 15: S_123456_Administrative_Error_HIH_Status_Response.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<ns0:SubmitPADeterminationResponseResult
xmlns:ns0="http://esmd.ois.cms.hhs.gov/v1/rc" returnCode="1"
serviceSuccessful="true">
    <statusDescription>Sent administrative error response
delivery to HIH</statusDescription>
    <ESMDTransaction TransactionId="123456" DeliveryType="S"/>
</ns0:SubmitPADeterminationResponseResult>
```

11.1.8 Administrative Error Response Validation Error

When the RC Client sends an Administrative Error Response to esMD, the esMD application processes and sends the Administrative Error Response to the HIH. If there is an error in processing the Administrative Error Response submitted by the RC, the esMD application generates the Administrative Error Response Validation Error, as detailed in Table 16: M_123456_Administrative_Response_Validation_Error.xml, and sends it to the RC. The RC will correct the administrative error response and resubmits it. Refer to the code located in Table 16: M_123456_Administrative_Response_Validation_Error.xml.

Table 16: M_123456_Administrative_Response_Validation_Error.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<ns0:SubmitPADeterminationResponseResult
xmlns:ns0="http://esmd.ois.cms.hhs.gov/v1/rc" returnCode="1"
serviceSuccessful="true">
    <statusDescription>ESMD internal system error (validation
failure) for administrative error response, please
resubmit</statusDescription>
    <ESMDTransaction TransactionId="123456" DeliveryType="M"/>
</ns0:SubmitPADeterminationResponseResult>
```

11.1.9 Virus Scan Error Response

When the RC Client sends any outbound file to esMD, the esMD application sends it to the Virus Scan Gateway for virus scan. If there are any viruses detected in the file received, the esMD application sends the message detailed in Table 17: X_123456_Virus_Scan_Error.xml to the RC. The RC Client will then pull this Virus Scan Error, stop the inbound and outbound processes, and lock down the RC Client to prevent RC Client from interacting with esMD. In this situation, the RC Client does not enable recovery, and the RC will contact esMD Help Desk. Refer to the code located in Table 17: X_123456_Virus_Scan_Error.xml.

Table 17: X_123456_Virus_Scan_Error.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:RCPickupNotificationResponse
xmlns:tns="http://esmd.ois.cms.hhs.gov/v1/rc/config"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config esmd-
config.xsd">
  <ESMDTransactionId>123456</ESMDTransactionId>
  <ErrorInfo>
    <ErrorCode>560</ErrorCode>
    <ErrorName>VirusFound</ErrorName>
    <ErrorDescription>ESMD validation error:  Submission is infected with
virus</ErrorDescription>
  </ErrorInfo>
  <Status>FAILED</Status>
  <StatusDesc>Outbound Response File contains virus and so the response is
rejected.</StatusDesc>
</tns:RCPickupNotificationResponse>
```

11.1.10 PA Review Result HIH Status Response

When the RC Client sends a PA Review Result to esMD, the esMD application processes the file and sends the PA Review Result to the HIH. The esMD application submits the PA Review Result HIH Status Response, detailed in Table 18:

N_123456_PA_Review_Result_HIH_Status_Response.xml, and sends it to the RC, indicating the result was sent to the HIH. Please refer to the code located in Table 13: A_123456_Pickup_HIH_Status_Response.xml.

Table 18: N_123456_PA_Review_Result_HIH_Status_Response.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<64ool:SubmitPADeterminationResponseResult
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc
../../config/esmd-rc.xsd "
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:esmd1="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
xmlns:64ool="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <statusDescription>PA Review Decision Response results -
Successfully delivered to HIH</statusDescription>
    <ESMDTransaction DeliveryType="N" TransactionId="123456"/>
</64ool:SubmitPADeterminationResponseResult>
```

11.1.11 PA Review Result Validation Error Response

When the RC Client sends a PA Review Result to esMD, the esMD application processes and sends the PA Review Result to the HIH. If there is an error in processing the PA Review Result submitted by the RC, the esMD application generates the PA Results Response Error, as detailed in Table 19: V_123456_PA_Review_Response_Validation_Error.xml, and sends it to the RC. The RC will correct the response and resubmits the PA Results Result. Refer to the code located in Table 19: V_123456_PA_Review_Response_Validation_Error.xml.

Table 19: V_123456_PA_Review_Response_Validation_Error.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<64ool:SubmitPADeterminationResponseResult
xsi:schemaLocation="http://esmd.ois.cms.hhs.gov/v1/rc/config/es
md-rc.xsd "
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:esmd1="http://esmd.ois.cms.hhs.gov/v1/rc/transaction"
xmlns:64ool="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:cmsbt="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <statusDescription>statusDescription</statusDescription>
    <ESMDTransaction DeliveryType="V" TransactionId="123456"/>
    <ValidationFailure>
        <FailureCode>556</FailureCode>
        <FailureReason>ESMD validation error: Decision
Indicator must be A, N, or M</FailureReason>
```

```
    </ValidationFailure>
</65ool:SubmitPADeterminationResponse>
```

11.2 Outbound

Note for API users: Please refer to the properties files packaged with the source code for more details on the reference data needed to populate the outbound XMLs described in this section.

The RC Client transfers the following messages during the outbound process:

- Pickup Notification;
- Error Pickup Notification;
- Review Decision Response to PA Request;
- Error Response to PA request; and
- Administrative Error Response to Inbound Submissions.

11.2.1 Pickup Notification

The RC Client generates pickup notifications for all inbound files with delivery type "E" pulled from TIBCO MFT server and processed successfully, as detailed in Table 20: P_186303_Pickup_Notification.xml. Refer to the code located in Table 20: P_186303_Pickup_Notification.xml.

Table 20: P_186303_Pickup_Notification.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:RCPickupNotification
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config">
    <ESMDTransactionId>186303</ESMDTransactionId>
    <RoutingId>ES9999</RoutingId>
    <PickupTime>2015-03-18T14:42:35.292-04:00</PickupTime>
    <SubmissionTime>2015-03-12T22:51:09.632-
04:00</SubmissionTime>
</ns2:RCPickupNotification>
```

11.2.2 Error Pickup Notification

The RC Client generates pickup error notifications for all inbound files pulled from TIBCO MFT and processed unsuccessfully, as detailed in Table 21: P_186303_Pickup_Error_Notification.xml. The processing errors are generated in two scenarios:

1. Checksum verification failed (i.e., the payload file received by the RC client does not match the file sent by esMD); and

2. Extraction was unsuccessful (i.e., the RC client could not successfully unzip the file received from the server).

Refer to the code located in Table 21: P_186303_Pickup_Error_Notification.xml.

Refer to the Section 11.2.5 Administrative Error Response to Inbound Submissions for more details on the errors sent from RC to the esMD for an inbound submission processing error.

Table 21: P_186303_Pickup_Error_Notification.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:RCPickupNotification
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc/config">
    <ESMDTransactionId>186303</ESMDTransactionId>
    <RoutingId>ES9999</RoutingId>
    <PickupTime>2015-03-18T14:42:35.292-04:00</PickupTime>
    <SubmissionTime>2015-03-12T22:51:09.632-
04:00</SubmissionTime>
    <ErrorInfo>
        <ErrorCode>535</ErrorCode>
       <ErrorName> ERROR VERIFYING PAYLOAD CHECKSUM </ErrorName>
       <ErrorDescription> ESMD_535 - RC Client processing error (Checksum
issue).  Please resubmit.</ErrorDescription>
    </ErrorInfo>
</ns2:RCPickupNotification>
```

11.2.3 Review Decision Response to PA Request

The Review Decision Response Result to PA Request is the XML message from the RC to the HIH to inform the HIH of the review decision.

11.2.3.1 PA Review Response with Decision Indicator 'Affirmed'

Table 22: E_1523121_PA_Review_Response.xml shows the XML message generated for a review response with decision indicator as "Affirmed".

The RC process for entering an A-Affirmed decision, which creates the XML indicated below, is detailed in Section 4.1 How to Enter an A-Affirmed Decision.

Table 22: E_1523121_PA_Review_Response.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:SubmitPADeterminationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:ns3="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <ESMDTransaction RoutingId="ES9999" DeliveryType="E"
TransactionId="1523121"/>
```

```
    <PAReviewResponse>
        <CreationTime>2015-03-17T23:06:27.637-
04:00</CreationTime>
        <SubmissionTime>2015-03-17T23:07:28.048-
04:00</SubmissionTime>
        <ProcedureLevelRecordList>
            <ProcedureLevelRecord>
                <ProcedureCode>K0802</ProcedureCode>
                <DecisionIndicator>A</DecisionIndicator>

<UniqueTrackingNumber>A0014280106700</UniqueTrackingNumber>
            </ProcedureLevelRecord>
        </ProcedureLevelRecordList>
    </PAReviewResponse>
</ns2:SubmitPADeterminationResponse>
```

11.2.3.2 PA Review Response with Decision Indicator "Affirmed" with Change

Table 23: E_1523124_PA_Review_Response.xml shows the XML message generated for a review response with decision indicator as 'Affirmed' with changes.

The RC process for entering an M-Affirmed with Change decision, which creates the XML indicated below, is detailed in the Section 4.2 How to Enter a M-Affirmed with Change Decision.

Please refer to the Appendix D: PA Reason Codes for more information on the reason codes used in the PA Review Response.

Table 23: E_1523124_PA_Review_Response.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:SubmitPADeterminationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:ns3="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <ESMDTransaction RoutingId="ES9999" DeliveryType="E"
TransactionId="1523124"/>
    <PAReviewResponse>
        <CreationTime>2015-03-17T23:03:29.861-
04:00</CreationTime>
        <SubmissionTime>2015-03-17T23:05:18.938-
04:00</SubmissionTime>
        <ProcedureLevelRecordList>
            <ProcedureLevelRecord>
                <ProcedureCode>A0426</ProcedureCode>
                <DecisionIndicator>M</DecisionIndicator>
                <UnitsDateRangeRecord>
                    <ApprovedUnits>9</ApprovedUnits>
```

```
                    <ApprovedDateRange>
                        <StartDate>2015-03-18</StartDate>
                        <EndDate>2015-03-26</EndDate>
                    </ApprovedDateRange>
                </UnitsDateRangeRecord>

<UniqueTrackingNumber>A0014280106621</UniqueTrackingNumber>
                <ReasonCodeList>
                    <ReasonCode>ADB23</ReasonCode>
                    <ReasonCode>WVYZ1</ReasonCode>
                </ReasonCodeList>
            </ProcedureLevelRecord>
        </ProcedureLevelRecordList>
    </PAReviewResponse>
</ns2:SubmitPADeterminationResponse>
```


11.2.3.3 PA Review Response with Decision Indicator "Non Affirmed"

Table 24: E_1541233_PA_Review_Response.xml shows the XML message generated for a review response with decision indicator as "Non Affirmed".

The RC process for entering an N-Non Affirmed decision, which creates the XML indicated below, is detailed in the Section 4.3 How to Enter a N-Non Affirmed Decision.

Please refer to the Appendix D: PA Reason Codes for more information on the reason codes used in the PA Review Response.

Please refer to the Appendix C: Industry Codes for more details on the Industry Codes to be used in an N-Non Affirmed response.

Note for API users: For the XDR PA Responses, the IndustryCodeRecordList will be discarded if RC sends the list, when the esMD system processes the PA Review Response.

Table 24: E_1541233_PA_Review_Response.xml


```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:SubmitPADeterminationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:ns3="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <ESMDTransaction RoutingId="ES9999" DeliveryType="E"
TransactionId="1541233"/>
    <PAReviewResponse>
        <CreationTime>2015-03-17T23:07:32.455-
04:00</CreationTime>
        <SubmissionTime>2015-03-17T23:08:40.451-
04:00</SubmissionTime>
```

```
        <ProcedureLevelRecordList>
            <ProcedureLevelRecord>
                <ProcedureCode>K0802</ProcedureCode>
                <DecisionIndicator>N</DecisionIndicator>

<UniqueTrackingNumber>A0014280106680</UniqueTrackingNumber>
                <IndustryCodeRecordList>
                    <IndustryCodeRecord>
                        <IndustryCode>04</IndustryCode>
                        <IndustryCodeDescription>Authorized
Quantity Exceeded</IndustryCodeDescription>
                    </IndustryCodeRecord>
                    <IndustryCodeRecord>
                        <IndustryCode>16</IndustryCode>
                        <IndustryCodeDescription>Inappropriate
facility type</IndustryCodeDescription>
                    </IndustryCodeRecord>
                </IndustryCodeRecordList>
                <ReasonCodeList>
                    <ReasonCode>PMD1A</ReasonCode>
                    <ReasonCode>PMD3A</ReasonCode>
                    <ReasonCode>PMD2B</ReasonCode>
                </ReasonCodeList>
            </ProcedureLevelRecord>
        </ProcedureLevelRecordList>
    </PAReviewResponse>
</ns2:SubmitPADeterminationResponse>
```

11.2.4 Error Response to PA request

The Error Response to PA Request is the XML message from the RC to the HIH, to inform the HIH of the error response as detailed in Table 25: E_1521342_PA_Review_Response.xml.

Please refer to the Appendix B: Reject Error Codes for more information on the error codes used in the Error Review Response for a PA Request.

Table 25: E_1521342_PA_Review_Response.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:SubmitPADeterminationResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:ns3="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <ESMDTransaction RoutingId="ES9999" DeliveryType="E"
TransactionId="1521342"/>
    <PAReviewResponse>
```

```
        <CreationTime>2015-03-17T23:10:22.327-
04:00</CreationTime>
        <SubmissionTime>2015-03-17T23:11:46.176-
04:00</SubmissionTime>
        <ErrorResponseDetail>
            <DecisionIndicator>R</DecisionIndicator>
            <RejectErrorCodeRecordList>
                <RejectErrorCodeRecord>
                    <ErrorCategoryName>Medical-
Info</ErrorCategoryName>
                    <ErrorCodeRecordList>
                        <ErrorCodeRecord>
                            <ErrorCode>15</ErrorCode>
                            <ErrorCodeDescription>Number of
Units is missing or invalid</ErrorCodeDescription>
                        </ErrorCodeRecord>
                        <ErrorCodeRecord>
                            <ErrorCode>33</ErrorCode>
                            <ErrorCodeDescription>Place of
Service code is missing or invalid</ErrorCodeDescription>
                        </ErrorCodeRecord>
                    </ErrorCodeRecordList>
                </RejectErrorCodeRecord>
                <RejectErrorCodeRecord>

<ErrorCategoryName>Beneficiary</ErrorCategoryName>
                    <ErrorCodeRecordList>
                        <ErrorCodeRecord>
                            <ErrorCode>64</ErrorCode>
                            <ErrorCodeDescription>HIC Number is
missing or invalid</ErrorCodeDescription>
                        </ErrorCodeRecord>
                    </ErrorCodeRecordList>
                </RejectErrorCodeRecord>
            </RejectErrorCodeRecordList>
            <ReasonCodeList>
                <ReasonCode>PMD1A</ReasonCode>
                <ReasonCode>PMD2C</ReasonCode>
                <ReasonCode>PMD3A</ReasonCode>
                <ReasonCode>PMD3B</ReasonCode>
            </ReasonCodeList>

<UniqueTrackingNumber>A0014280106600</UniqueTrackingNumber>
        </ErrorResponseDetail>
    </PAReviewResponse>
</ns2:SubmitPADeterminationResponse>
```

11.2.5 Administrative Error Response to Inbound Submissions

The Administrative Error Response is the XML message from the RC to the HIH to inform the HIH of the administrative error response to inbound submissions.

Note: Section 6 How to Submit an Inbound Submission Error on the Administrative Error Response to Inbound Submissions Tab describes the process of creating an XML message, using the RC Client.

Table 26: D_1532432_Administrative_Error_Response.xml provides the PA Review Decision Response XML.

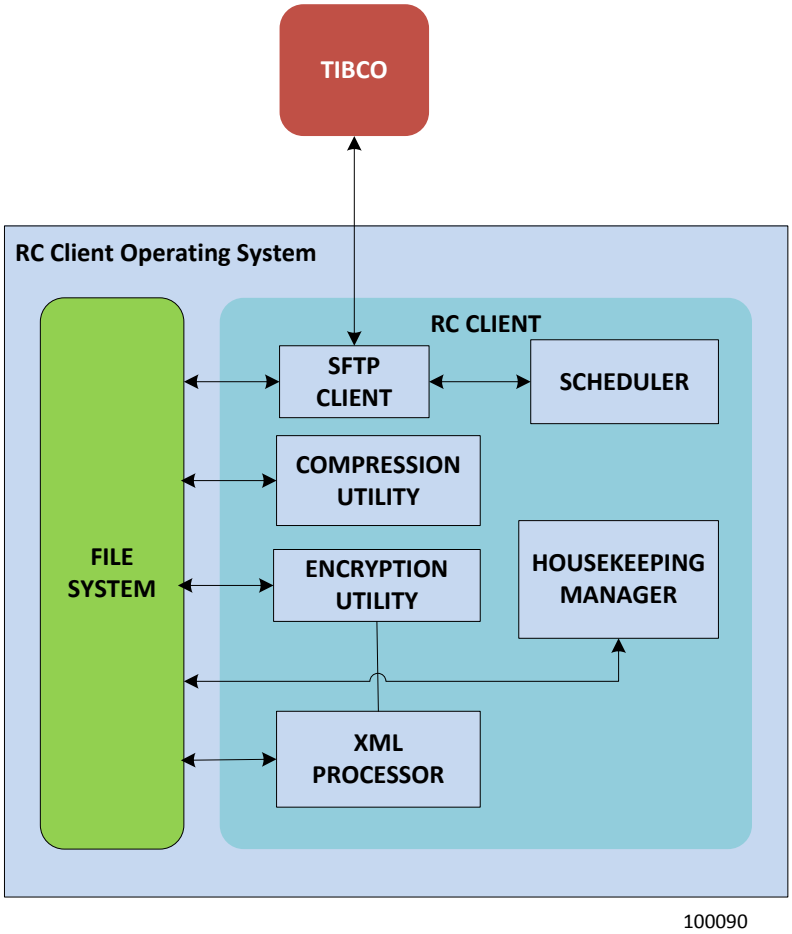
Table 26: D_1532432_Administrative_Error_Response.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ns2:SubmitAdministrativeErrorResponse
xmlns:ns2="http://esmd.ois.cms.hhs.gov/v1/rc"
xmlns:ns3="http://esmd.ois.cms.hhs.gov/v1/rc/cmsbt">
    <ESMDTransaction RoutingId="ES9999" DeliveryType="D"
TransactionId="1532432"/>
    <AdministrativeErrorResponse>
        <CreationTime>2015-03-17T23:11:57.521-
04:00</CreationTime>
        <SubmissionTime>2015-03-17T23:12:10.931-
04:00</SubmissionTime>
        <ErrorResponseList>
            <ErrorResponseRecord>
                <ErrorCode>ESMD_410</ErrorCode>
                <ErrorName>Cannot Read Files / Corrupt
Files</ErrorName>
            </ErrorResponseRecord>
        </ErrorResponseList>
    </AdministrativeErrorResponse>
</ns2:SubmitAdministrativeErrorResponse>
```

12. RC Client Components

Figure 7: RC Client Components shows the internal components of RC Client application. The following sections describe each component in detail.

Figure 7: RC Client Components



12.1 SFTP Client

The SFTP Client is an internal component of the RC Client. It provides the following functionality:

- Connects to the TIBCO MFT server using EIDM ID;
- Lists the available documents on the TIBCO MFT server;
- Pulls the documents to the RC Client; and
- Pushes the outbound documents from RC Client to the TIBCO MFT server.

12.2 Compression Utility

The Compression utility allows the RC Client to extract the payload, metadata file, and messages from the compressed file downloaded from the TIBCO MFT server. The RC Client uses the zip file format.

The same utility is used to create compressed file logs for extraction.

12.3 Encryption Utility

The Encryption utility encrypts the login credentials that will be stored in memory for the duration of the RC Client program execution. The Encryption utility is described in detail in Section 15.1 Security.

12.4 XML Processor

The XML Processor supports creating XML messages to send to esMD as well as loading the configuration files for the RC Client.

12.5 Scheduler

After the RC Client starts, the polling cycle begins. The poll is a redundant cycle; you can configure the interval (for example, 1 hour or 4 hours) through the RC Client property file. The Schedule component controls the RC Client threads and ensures the RC Client runs in regular intervals determined by the “checkFrequency” parameter in the XML Configuration File.

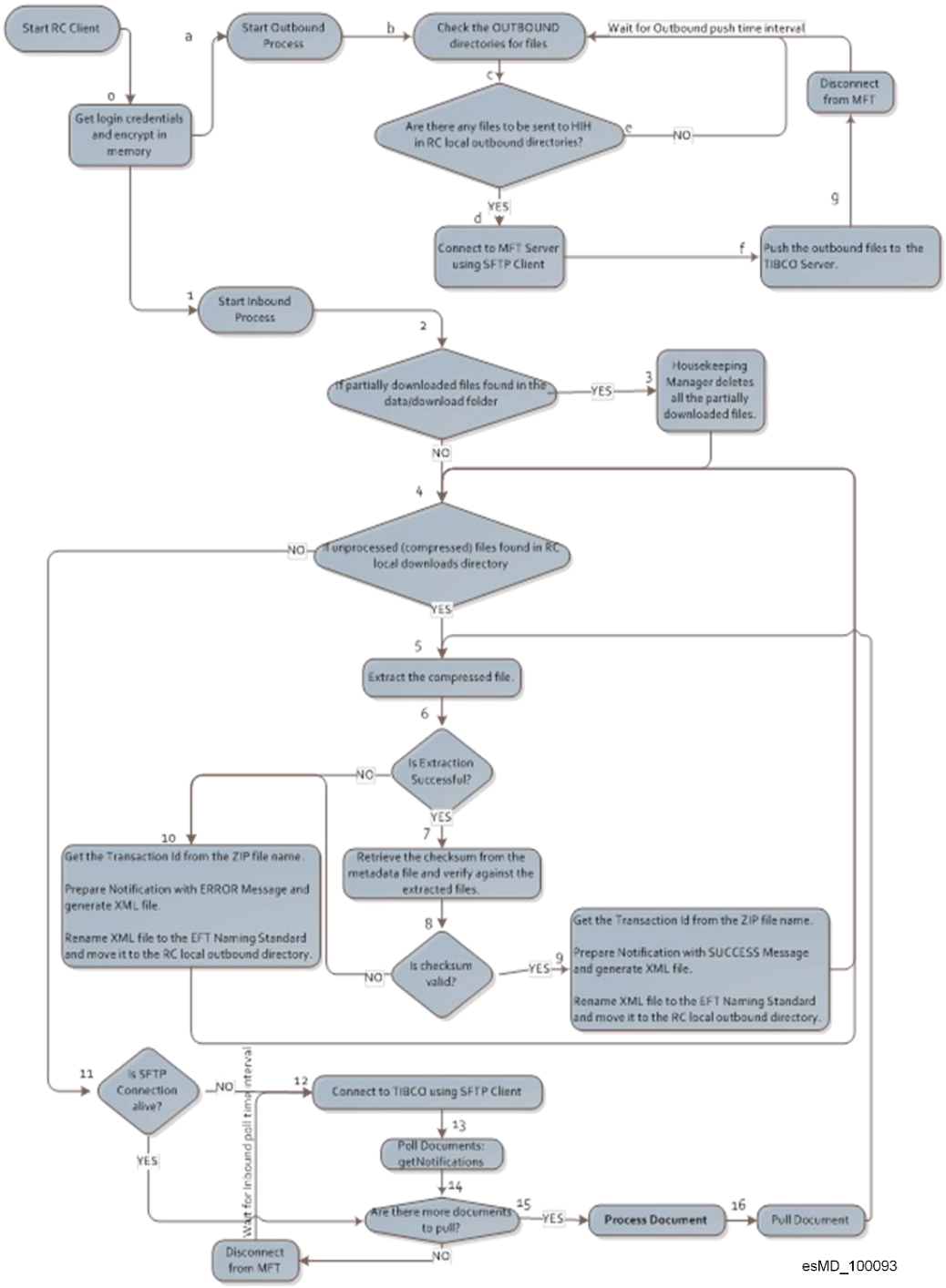
12.6 Housekeeping Manager

The Housekeeping Manager allows the RC Client to recover from any abnormal terminations with the exception of a Virus lockdown. In this situation, the RC Client does not enable recovery, and the RC must contact the esMD Help Desk.

13. RC Client Workflow

The workflow associated with Figure 7: RC Client Components is broken down in Figure 8: RC Client Workflow, followed by a detailed description of the workflow.

Figure 8: RC Client Workflow



13.1 Start RC Client

The RC Client starts on the RC machine or server. It loads the XML Configuration File.

13.1.1 Login and Encryption

The RC Client prompts the user for the following details:

1. EIDM User ID; and
2. EIDM Password.

After successful login, TIBCO login credentials are encrypted in memory and used when needed to log in to the TIBCO MFT server. The RC Client initiates two threads, one for the inbound process and one for the outbound process in sections 13.2 Outbound Process and 13.3 Inbound Processes, respectively. These processes are described in the next section.

13.2 Outbound Process

13.2.1 Outbound Start

The RC Client loads configuration parameters for the outbound process from the XML configuration file. The configuration parameters are as follows:

- Directories used by the RC Client to create the outbound files (`outputDirectory`);
- The remote outbound directory to push the files to (`remoteOutboundDir`);
- Push frequency (`pushFrequency`);
- The outbound file name prefix for the TIBCO MFT server (`outboundFilePrefix`); and
- SFTP server details for the chosen environment (`ESMDSFTPServer`).

13.2.2 Get Outbound Documents

The RC Client checks the output directory for any files to be sent to the HIH. If any such files exist, the process continues to Step D (Connect); otherwise, the outbound process thread sleeps for the time interval determined by the pushFrequency parameter in the XML Configuration file.

13.2.3 Connect

The RC Client connects to the TIBCO MFT server using EIDM login credentials. The Encryption utility decrypts the credentials in memory and logs in to the TIBCO MFT server. If the user password is expired, the connection fails, prompting the user to provide the login information again.

13.2.4 Push

The RC Client pushes outbound files to the TIBCO MFT server. After that, the outbound process thread sleeps. The sleep time interval is determined by the outbound push frequency configuration parameter in the XML Configuration file.

13.3 Inbound Processes

13.3.1 Inbound Start

The RC Client loads Configuration parameters from the XML Configuration file. The configuration parameters are for the following inbound processes:

- Pull frequency; and
- SFTP server details for the chosen environment.

13.3.2 Housekeeping

The Housekeeping Manager is responsible for the cleanup and recovery from any abnormal terminations. If the extraction process was interrupted during extraction in the previous run, then there will be compressed files in the local “temp” directory.

13.3.3 Extraction

The Housekeeping Manager extracts compressed files found in the local “temp” directory for the RC Client before it pulls any new documents from the TIBCO MFT server. It will extract the oldest files first. If the extraction is successful, RC Client proceeds to “checksum verification”; otherwise, RC Client creates an error pickup notification.

13.3.4 Checksum Verification

After the extraction is complete, the RC Client uses the XML Processor to parse the metadata file from the zip package. This metadata file contains the checksums for all payloads in the package. The RC Client verifies the checksum for each file in the package against the checksum in the metadata file. If the checksum is valid for all files, the RC Client will create a pickup notification; otherwise, the RC Client will create an error pickup notification.

13.4 Acknowledgements

13.4.1 Pickup Notification

If the RC Client successfully extracts and verifies compressed files, the RC Client sends a SUCCESS notification through esMD to inform the HIH that the document has been received and successfully processed.

To generate this SUCCESS notification, the RC Client should:

1. Get the Transaction ID from the compressed file name;
2. Prepare the notification with a SUCCESS message and generate an XML notification file; and
3. Rename the XML notification file to the EFT naming standard and move it to the outbound directory. Refer to Section 13.2 Outbound Process for more information.

13.4.2 Error Pickup Notification

If the RC Client encounters an error indicating a failure, while either extracting the compressed file or verifying the checksum for the contents of the package, the RC Client sends an error notification through the esMD system, asking the HIH to resubmit the package. In order to generate this error notification, the RC Client must:

- Obtain the Transaction Identifier (TID) from the compressed file name;
- Prepare the notification with an Error message;
- Generate an XML notification file; and
- Rename the XML notification file to the EFT naming standard and move it to the outbound directory. This file will be handled by the outbound process.

13.5 Connect

After the Housekeeping Manager completes preprocessing, the RC Client checks for an active connection to the TIBCO MFT server. If a connection is active, the RC Client uses this connection. If the connection is inactive, the RC Client uses the Encryption utility to decrypt the login credentials from memory and connects to the TIBCO MFT server.

13.6 Get Notifications

The RC Client uses the SFTP Client to get a list of the available inbound documents for the RC on the TIBCO MFT server.

13.7 Process Document

If any documents are available for the RC Client to pull from the TIBCO MFT server, the RC Client will go through the list to pull each document.

13.8 Pull Document

The RC Client uses the SFTP Client to pull each inbound document from the TIBCO MFT server. The RC Client then extracts the contents of the zip file and continues processing.

14. Release 4.0 Changes in the API

Table 27: Inbound Client Method lists the Inbound methods that was modified as part of Release 4.0 RC Client.

Table 27: Inbound Client Method

Method	Description
InboundProcessImpl.processMedicalDocumentation() .	• Extracts the zip file into the "download" directory using the extractDocument() method. • If extraction fails, calls the acknowledge method with an error event and exits. • After successful extraction, Verifies the extracted payloads, flat file rendering (FFR) and Cover sheet against the checksum in the metadata file using the checkPayloads() method. • If checksum fails, calls the acknowledge method with an error event. • If checksum passes, calls the acknowledge method with a success event. Routes the inbound files to the local directories

Table 28: Outbound Client Method (PA Review Decision) lists the Outbound methods in the that were modified as part of Release 4.0 RC Client.

Table 28: Outbound Client Method (PA Review Decision)

Method	Description
ESMDManualSubmitPAReviewResponseImpl. validationOfPAReviewDecisionResponse(PAReviewDecisionResponseBean paReviewDecisionResponseBean_)	• Gets the following information as input from PAReviewDecisionResponseBean: ○ Transaction ID; ○ Procedure Code; ○ Procedure Level Decision (Decision Indicator); ○ Number of Approved Units; ○ Approved Service Date; ○ Approved Service Date Range(Start Date and End Date); ○ Procedure Level UTN (Unique Tracking Number); ○ Reason Code(s); ○ Industry Code(s); ○ Creation Time; and ○ Submission Time. • Perform missing, format and length validations for all input data provided. ○ Creates the list of ValidationFailureBean Object for data which failed validation. ○ Builds the Message Object with status of validation result and also sets list of Validation Failure Bean object if there is any validation failure.
ESMDManualSubmitPAReviewResponseImpl createPAReviewResponseObject(PAReviewDecisionResponseBean paReviewDecisionResponseBean_)	• Gets the following information as input from PAReviewDecisionResponseBean: ○ Transaction ID; ○ Procedure Code; ○ Procedure Level Decision (Decision Indicator); ○ Number of Approved Units; ○ Approved Service Date; ○ Approved Service Date Range (Start Date and End Date); ○ Procedure Level UTN (Unique Tracking Number); ○ Reason Code(s); ○ Industry Code(s); ○ Creation Time; and ○ Submission Time. Sets Delivery Type as “E”. Builds the PA Review Response Type Object and sets it to SubmitPADeterminationResponse object.
ESMDManualSubmitPAReviewResponseImpl submitPAReviewDecisionResponse(PAReviewDecisionResponseBean paReviewDecisionResponseBean_)	• Takes the SubmitPADeterminationResponse object, creates the XML response file, compresses, and place it in the output directory. Pushes the response XML file to TIBCO MFT server by the outbound process.

Table 29: Outbound Client Method (PA Error (Rejected Decision) Response) lists Outbound – PA Error (Rejected Decision) Response methods that were modified as part of Release 4.0 RC Client.

Table 29: Outbound Client Method (PA Error (Rejected Decision) Response)

Method	Description
ESMDManualSubmitPAErrorResponseImpl. validationOfPAErrorResponse (PAErrorResponseBean paErrorResponseBean	• Gets the following information as input from PAErrorResponseBean: ○ Transaction ID; ○ Decision Indicator; ○ Request Level UTN (Unique Tracking Number); ○ Reason Code(s); ○ Reject Error Category(ies); ○ Reject Error Code(s); ○ Reject Error Description; ○ Creation Time; and ○ Submission Time. • Perform missing, length and format validations for all input data provided. • Creates the list of ValidationFailureBean Object for data which failed validation. • Builds the Message Object with status of validation result and also sets list of Validation Failure Bean object if there is any validation failure. • Returns the Message Object.
ESMDManualSubmitPAErrorResponseImpl.createPAErrorResponseObject(PAErrorResponseBean paErrorResponseBean_)	• Gets the following information as input from PAErrorResponseBean: ○ Transaction ID; ○ Decision Indicator; ○ Request Level UTN (Unique Tracking Number); ○ Reason Code(s); ○ Reject Error Category(ies); ○ Reject Error Code(s); ○ Reject Error Description; ○ Creation Time; and ○ Submission Time. • Sets Delivery Type as “E”. • Builds the PA Review Response Type Object and sets it to SubmitPADeterminationResponse object.
ESMDManualSubmitPAErrorResponseImpl .submitPAErrorResponse(PAErrorResponseBean paErrorResponseBean_)	• Takes the SubmitPADeterminationResponse object, creates the XML response file, compresses, and place it in the output directory. • Pushes the compressed response XML file to TIBCO MFT server by the outbound process.

Table 30: Outbound Client Methods (Administrative Error Response) lists Outbound – Administrative Error Response methods that were modified as part of Release 4.0 RC Client.

Table 30: Outbound Client Methods (Administrative Error Response)

Method	Description
ESMDManualSubmitAdminErrorResponseImpl. validationOfAdministrativeErrorResponse (AdministrativeErrorResponseBean administrativeErrorResponseBean_)	• Gets the following information as input from AdministrativeErrorResponseBean: ○ Transaction ID; ○ Administrative Error Code; ○ Administrative Error Name; ○ Creation Time; and ○ Submission Time. • Perform missing, format and length validations for all input data provided. • Creates the list of ValidationFailureBean Object for data which failed validation. • Builds the Message Object with status of validation result and also sets list of Validation Failure Bean object if there is any validation failure.
ESMDManualSubmitAdminErrorResponseImpl. createAdministrativeErrorResponseObject(AdministrativeErrorResponseBean administrativeErrorResponseBean_)	• Gets the following information as input from AdministrativeErrorResponseBean: ○ Transaction ID; ○ Administrative Error Code; ○ Administrative Error Name; ○ Creation Time; and ○ Submission Time. • Sets Delivery Type as “D” • Builds the Administrative Error Response Type Object and sets it to SubmitAdministrativeErrorResponse object.
ESMDManualSubmitAdminErrorResponseImpl. submitAdministrativeErrorResponse(AdministrativeErrorResponseBean administrativeErrorResponseBean_)	• Takes the SubmitAdministrativeErrorResponse object, creates the XML response file, compresses, and places it in the output directory. • Pushes the compressed response XML file to TIBCO MFT server by the outbound process.

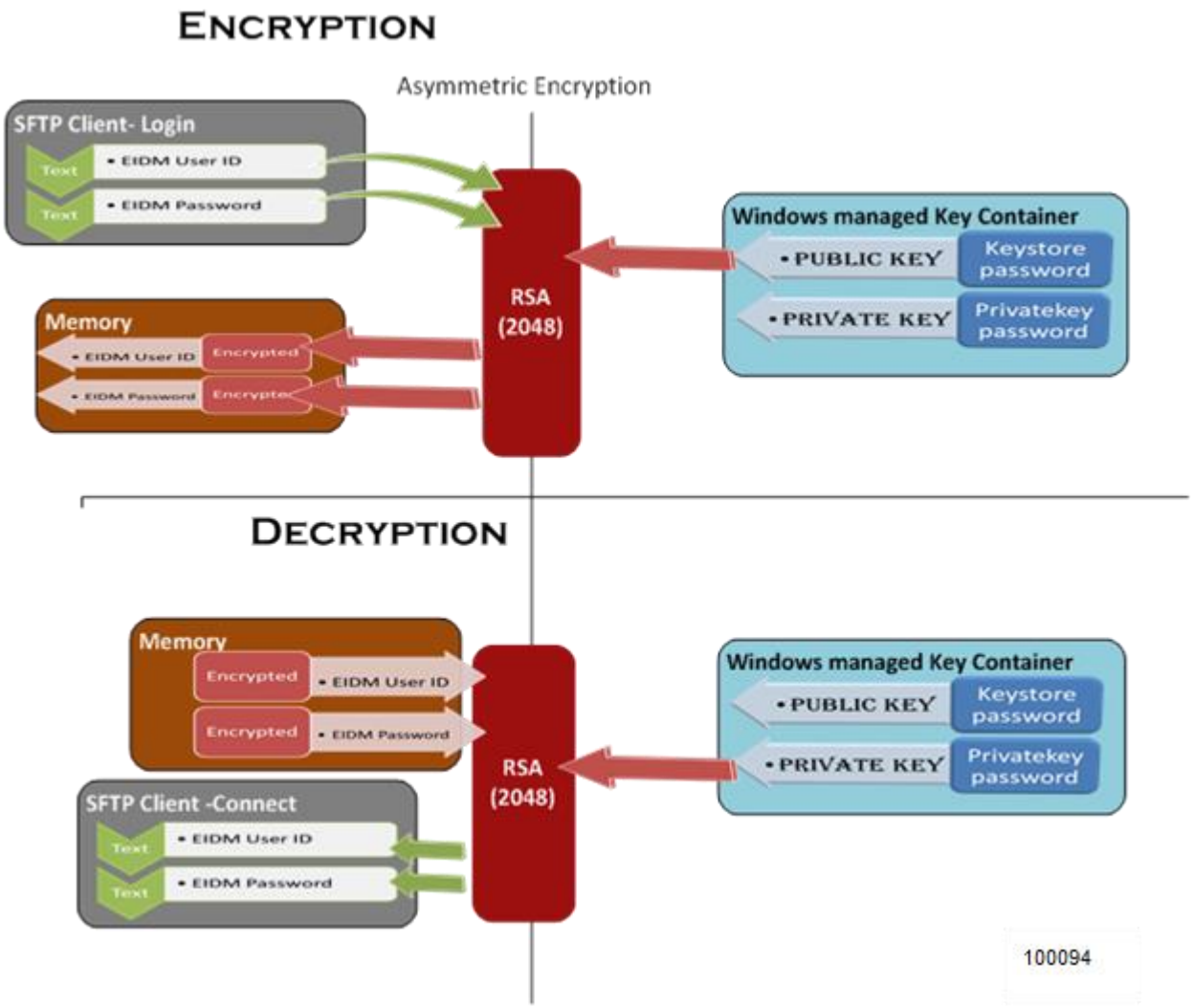
15. Java Client API

15.1 Security

When the RC Client starts, the user credentials are provided because they are stored in encrypted form in memory. Figure 9: Encryption and Decryption Process shows the processes used to safeguard the EIDM user credentials from exposure.

The RC Java Client uses RSA asymmetric encryption algorithms to secure the login credentials.

Figure 9: Encryption and Decryption Process



15.2 Java API Documentation

This section discusses API methods that can be called for a custom solution to interface with the TIBCO MFT server. If you, as the RC, choose to use the RC Java client out-of-the-box, skip this section.

15.2.1 Login

Table 31: Login Details lists the methods and their descriptions used in the login process.

Table 31: Login Details

No.	Method	Description
1.	public LoginDetails loginAndEncrypt(SftpDetails sftpDetails_, ESMDConfig.KeyStoreInfo keyStoreInfo_, LoginBean loginBean_) throws Exception;	Logs into the server and stores the encrypted login information. Parameters: 1. sftpDetails_ – The SFTP server Details; 2. keyStoreInfo_ – The Keystore Details; and 3. loginBean_ – The Login Details (User ID and Password). Returns: The LoginDetails Object with the following properties populated: 1. encryptedUID – Encrypted User ID; 2. encryptedPWD – Encrypted Password; and 3. message – status(TRUE/FALSE) and description if any exceptions occurred.
2.	public LoginDetails decryptAndLogin(LoginDetails loginDetails_, SftpDetails sftpDetails_, ESMDConfig.KeyStoreInfo keyStoreInfo_) throws Exception;	Decrypts the login credentials passed in the LoginDetails object and logs into the TIBCO MFT server. Parameters: 1. loginDetails_ – the LoginDetails object with the following properties populated: • encryptedUID – Encrypted User ID; and • encryptedPWD – Encrypted Password; 2. sftpDetails_ – The SFTP server Details; and 3. keyStoreInfo_ – The Keystore Details. Returns: The LoginDetails Object with the following properties populated: 1. encryptedUID – Encrypted User ID; 2. encryptedPWD – Encrypted Password; and 3. channelSftp – SFTP Channel connected.

15.2.2 Inbound

Table 32: Inbound Method Details lists the methods and their descriptions used in the inbound process.

Table 32: Inbound Method Details

No.	Method	Description
1.	public List<String> getNotifications(LoginDetails loginDetails_,String remoteDownloadDirectoryPath_, String filePattern_) throws SftpException ;	Uses the LoginDetails object to list the remote directory. Parameters: 1. loginDetails_ – the LoginDetails object with the following properties populated: • encryptedUID – Encrypted User ID; • encryptedPWD – Encrypted Password; and • channelSftp – SFTP Channel connected; 2. remoteDownloadDirectoryPath_– The remote directory path to download from as a String; and 3. filePattern_ – The File Name Pattern to look for as a String. Returns: The List<String> with the filenames to pull.
2.	public void pullDocument(String remoteDocumentName_, String localDocumentName_, LoginDetails loginDetails_) throws Exception;	1. Pulls the document (namely, the zip file) from the TIBCO MFT server with the name remoteDocumentName; and 2. Saves it as localDocumentName_ using the loginDetails _ to pull the file from TIBCO MFT server. Parameters: 1. remoteDocumentName_ – The remote file to pull as a String; 2. localDocumentName_ – The local file name to save as a String; and 3. loginDetails_ - the LoginDetails object with the following properties populated: • encryptedUID – Encrypted User ID; • encryptedPWD – Encrypted Password; and • channelSftp – SFTP Channel connected.
3.	public String extractDocument(File localDocumentName_, File localTargetDirectory_) throws Exception;	Extracts the zip file downloaded from the TIBCO MFT server. Parameters: 1. localDocumentName_ - The local zip file to extract; and 2. localTargetDirectory _ - The target directory to place the extracted contents. Returns: The extracted Directory name as a String.

No.	Method	Description
4.	public boolean processMedicalDocumentation(String remoteDocumentName_);	This method does the following: 1. Extracts the zip file into the “download” directory using the extractDocument() method; 2. If extraction fails, calls the acknowledge method with an error event; 3. After successful extraction, verifies the extracted payloads, flat file rendering file and cover sheet against the checksum in the metadata file using the checkPayloads() method; 4. If checksum fails, calls the acknowledge method with an error event; and 5. If checksum passes, calls the acknowledge() method with a success event. Parameter: 1. localDocumentPath_ - The local document name to process. Returns: The Boolean status of the processing for that document.
5.	public String acknowledge(RCPickupNotification rcPickupNotification_) throws Exception;	Generates the pickup notification for a downloaded document. If the ErrorInfo object is populated, it generates an error pickup notification. If the ErrorInfo object is null, it generates a pickup notification. Parameter: 1. rcPickupNotification_ - The RCPickupNotification object. Returns: The TIBCO MFT server ready compressed file name created in the output directory as a String.
6.	public boolean checkPayloads(File localExtractedDirectory_, RetrieveMedicalDocumentationResponse retrieveMedicalDocumentationResponse_);	Checks the payload, FFR, and coversheet files against the metadata from the package. Parameters: 1. localExtractedDirectory_ – The directory in which the payloads, ffr and coversheet were extracted to as a File; and 2. retrieveMedicalDocumentationResponse_ – The metadata xml as object. Returns: The status of the checksum verification.

15.2.3 Outbound

Table 33: Retrieval of Outbound Documents Details provides the retrieval of Outbound documents.

Table 33: Retrieval of Outbound Documents Details

No.	Method	Description
1.	public List<String> getOutboundDocuments(String localOutputDirectoryPath_, String localOutboundDocumentNamePattern_) throws Exception;	This method is used to retrieve the list of outbound documents in the "output" directory to push. Parameters: 1. localOutputDirectoryPath_ – The local "output" directory to push files from as a String; and 2. localOutboundDocumentNamePattern_ – The file name pattern to push as a String. Returns: The List<String> with the names of the Outbound files in the "output" directory.
2.	public void pushDocument(String localOutboundDocumentPath_, String remoteOutboundDirectoryName_, LoginDetails loginDetails_) throws Exception;	This method is used to push a local compressed document from the "output" directory to the TIBCO MFT server. Parameters: 1. localOutboundDocumentPath_ – The name of the file to push as a String; 2. remoteOutboundDirectoryName_ – The remote directory name to push to as a String; and 3. loginDetails_ - The LoginDetails object with the following properties populated: • encryptedUID – Encrypted User ID; • encryptedPWD – Encrypted Password; and • channelSftp – SFTP Channel connected.

15.2.4 PA Review Decision Response

Table 34: Manual Submission of PA Review Decision Response details the methods to submit the PA Review Decision (Affirmed, Affirmed with Change and Non Affirmed) Response.

Table 34: Manual Submission of PA Review Decision Response

No.	Method	Description
1.	public Message validationOfPAReviewDecisionResponse(PAReviewDecisionResponseBean paReviewDecisionResponseBean_);	This method takes PAReviewDecisionResponseBean object as input, which has the review decision response information provided by user and validates all information before generating the XML response. Parameter: 1. paReviewDecisionResponseBean_ – The PAReviewDecisionResponseBean object to use. Returns: The Message Object which has status of validations result and also the list of Validation Failure Bean object if there is any validation failure with the data provided by the user.

No.	Method	Description
2.	public SubmitPADeterminationResponse createPAReviewResponseObject(PAReviewDecisionResponseBean paReviewDecisionResponseBean_) throws Exception;	This method takes PAReviewDecisionResponseBean object as input which has the review decision response information and creates the SubmitPADeterminationResponse object. Parameter: 1. paReviewDecisionResponseBean_ – The PAReviewDecisionResponseBean object to use. Returns: The SubmitPADeterminationResponse object populated with the data provided by the user.TIBCO MFT server.
3.	public String createCompressedTIBCOFileForPAReviewDecisionResponse(SubmitPADeterminationResponse submitPADeterminationResponse_) throws Exception;	This method is used create the XML file and compress it into a TIBCO MFT server file. Parameter: 1. submitPADeterminationResponse_ – The SubmitPADeterminationResponse object to use. Returns: The compressed outbound file name ready to be pushed by the outbound process.

15.2.5 PA Error (Rejected Decision) Response

Table 35: Manual Submission of PA Error (Rejected Decision) Response details the methods to submit the PA Error (Rejected Decision) Response.

Table 35: Manual Submission of PA Error (Rejected Decision) Response

No.	Methods	Description
1.	public Message validationOfPAErrorResponse (PAErrorResponseBean paErrorResponseBean_);	This method takes PAErrorResponseBean object as input which has the review error (rejected decision) response information provided by user and validates all that information before generating the response XML. Parameter: 1. paErrorResponseBean_ – The PAErrorResponseBean object to use. Returns: The Message Object which has status of validations result and also the list of Validation Failure Bean object if there is any validation failure with the data provided by the user.
2.	public SubmitPADeterminationResponse createPAErrorResponseObject(PAErrorResponseBean paErrorResponseBean_) throws Exception;	This method takes PAErrorResponseBean object as input which has the review error (rejected decision) response information provided by user and creates the SubmitPADeterminationResponse object. Parameter: 1. paErrorResponseBean_ – The PAErrorResponseBean object to use. Returns: The SubmitPADeterminationResponse object populated with the data provided by the user.

No.	Methods	Description
3.	public String createCompressedTIBCOFileForPAErrorResponse(SubmitPADeterminationResponse submitPADeterminationResponse_) throws Exception;	This method is used create the XML file and compress it into a TIBCO MFT server file. Parameter: 1. submitPADeterminationResponse_ – The SubmitPADeterminationResponse object to use. Returns: The compressed outbound file name ready to be pushed by the outbound process.

15.2.6 Administrative Error Response to Inbound Submissions

Table 36: Manual Submission of Administrative Error Response details the methods to submit the Administrative Error Response to an Inbound submission.

Table 36: Manual Submission of Administrative Error Response

#	Methods	Description
1.	public Message validationOfAdministrativeErrorResponse (AdministrativeErrorResponseBean administrativeErrorResponseBean_);	This method takes AdministrativeErrorResponseBean object as input which has the administrative error response information provided by user and validates all that information before generating the response XML. Parameter: 1. administrativeErrorResponseBean_ – The AdministrativeErrorResponseBean object to use. Returns: The Message Object which has status of validations result and also the list of Validation Failure Bean object if there is any validation failure with the data provided by the user.
2.	private SubmitAdministrativeErrorResponse createAdministrativeErrorResponseObject(AdministrativeErrorResponseBean administrativeErrorResponseBean_) throws Exception;	This method takes AdministrativeErrorResponseBean object as input which has the administrative error response information provided by user and creates the SubmitAdministrativeErrorResponse object. Parameter: 1. administrativeErrorResponseBean_ – The AdministrativeErrorResponseBean object to use. Returns: The SubmitAdministrativeErrorResponse object populated with the data provided by the user.
3.	public String createCompressedTIBCOFileForAdministrativeErrorResponse(SubmitAdministrativeErrorResponse submitAdministrativeErrorResponse_) throws Exception;	This method is used create the XML file and compress it into a TIBCO MFT server file. Parameter: 1. submitAdministrativeErrorResponse_ – The SubmitAdministrativeErrorResponse object to use. Returns: The compressed outbound file name ready to be pushed by the outbound process.

15.2.7 Utilities - Encryption

Note: The Java Client release from April 28, 2014 does not include the encryption of login credentials. This section depicts the planned design and is subject to change. This guide will be

updated as required when the security implementation is completed.

Table 37: Encryption provides the details on the EMSD.RcClient.Encryption.EncryptionUtil methods.

Table 37: Encryption

#	Methods	Description
1.	public String encryptKSPassword(String keyStorePassword_) throws Exception;	This method encrypts the Keystore password so it can be stored in the configuration file. Parameter: 1. keyStorePassword_– The password to encrypt as a String. Returns: The Encrypted Keystore Password using "PBEWithMD5AndTripleDES.
2.	public String encryptPKPassword(String privateKeyPassword_) throws Exception;	This method encrypts the Private Key password so it can be stored in the configuration file. Parameter: 1. privateKeyPassword_ – The password to encrypt as a String. Returns: The Encrypted Private Key Password using "PBEWithMD5AndTripleDES".
3.	public Map<String, String> encryptCredentials(Map<String, String> loginInfo_, ESMDConfig.KeyStoreInfo keyStoreInfo_) throws Exception;	This method encrypts the EIDM login credentials using a RSA Public Key from the JKS Store. Parameters: 1. loginInfo_ - The Map<String, String> containing the UID and PWD as keys; and 2. keyStoreInfo_ - The ESMDConfig.KeyStoreInfo object with the following details populated: • keyStoreLocation – The JKS Store to use as a String; • encKeyInfo – The Encrypted Keystore password to load the JKS as a String; and • certAlias – The alias of the certificate to retrieve the public key as a String. Returns: The Map<String, String> of encrypted login credentials ENC_UID and ENC_PWD as keys.

#	Methods	Description
4.	public Map<String, String> decryptCredentials(Map<String, String> encryptedLoginInfo_, ESMDConfig.KeyStoreInfo keyStoreInfo_) throws Exception;	This method decrypts the EIDM login credentials using a RSA Private Key from the JKS Store. Parameters: 1. encryptedLoginInfo_ - The Map<String, String> of encrypted login credentials ENC_UID and ENC_PWD as keys; and 2. keyStoreInfo_ - The ESMDConfig.KeyStoreInfo object with the following details populated: • keyStoreLocation – The JKS Store to use as a String; • encKeyInfo - The Encrypted Keystore password to load the JKS as a String; • certAlias - The alias of the certificate to retrieve the public key as a String; and • encKeyInfoExt - The Encrypted private key password to load the private key from the JKS Store as a String. Returns: The Map<String, String> containing the UID and PWD as keys.

15.2.8 Test Connection

Refer to Table 38: Remote Troubleshooting for details on the ExecuteHandshake method.

Table 38: Remote Troubleshooting

#	Methods	Description
1.	public bool executeHandshake()	This sample method invokes a call to the TIBCO MFT server to pass login information to assist in remote troubleshooting. Returns: TRUE if handshake succeeded.

15.3 Logs

Table 39: RC Client Logs lists the logs the RC Client provides. The RC is advised to monitor the logs for errors and exceptions.

Table 39: RC Client Logs

Log	Description
config.log	Logging for the encryptConfig.bat utility.
handshake.log	Logging for the test connection process.
rc.log	Logging for the sample application.
Inbound.log	Logging for the Inbound Process.
outbound.log	Logging for the Outbound Process.
response.log	Logging for the Response File (PA Review and Administrative Error) Creation Process.

15.4 Utilities

Table 40: RC Client Utilities lists the utilities the RC Client provides.

Table 40: RC Client Utilities

Log	Description
encryptConfig.bat	Encrypts the provided passwords and updates the configuration XML file.
rcclient.bat	The RC Client User Interface Application. This application will have: 1) Login; 2) Review Decision Response to PA Request; 3) Error Response to PA Request; 4) Administrative Error Response to Inbound Submissions; and 5) Advanced/Debugging functionalities.

16. Error Codes

16.1 Errors: esMD to RC

Table 41: Error Codes Sent From esMD to RC lists all the error codes sent from the esMD to the RC.

Table 41: Error Codes Sent From esMD to RC

Error Code	Error Description
305	esMD validation error: Review Contractor PickUp Timestamp is not a valid Timestamp. Correct and resubmit.
306	esMD validation error: esMD Delivery Timestamp is not a valid Timestamp. Correct and resubmit.
539	esMD internal system error (Unzip failure). Resubmit.
544	esMD validation error: Reason Code is required when Decision Indicator is N or R. Correct and resubmit.
545	esMD validation error: Total number of Reason Codes cannot exceed 25. Reduce the number of Reason Codes and resubmit.
556	esMD validation error: Decision Indicator must be A, N, M or R. Correct and resubmit.
557	esMD validation error: Review Contractor Unique Tracking Number must be 1 - 50 alphanumeric characters with no special characters. Correct and resubmit.
558	esMD validation error: Reason Code does not exist in the esMD database. Correct and resubmit.
560	esMD validation error: Submission is infected with virus. Correct and resubmit.
562	esMD validation error: Unique Tracking Number is required when Decision Indicator is A, N, or M. Correct and resubmit.
565	esMD Internal System error: Unable to process your response. Correct and resubmit.
566	esMD validation error: A required element is either missing, has an invalid element format, or has an invalid length. Correct and resubmit.
567	esMD validation error: A Decision Indicator of 'M' is invalid for PMD PA response. Provide a valid Decision Indicator and resubmit.
569	esMD validation error: Number of Approved Units; and/or Approved Service Date or Approved Service Date Range are not required elements for this PA response and Procedure Level Decision. Correct and resubmit.
572	esMD validation error: Approved Service End Date is less than or equal to Approved Service Start Date. Correct and resubmit.
576	esMD validation error: Number of Approved Units, Approved Service Date, Approved Service Date Range, Industry Code(s) and Reason Code(s) are not allowed for this response. Correct and resubmit.
577	esMD validation error: Unable to parse response XML file. Correct XML and resubmit.
600	esMD validation error: Duplicate Reason Codes found. Correct and resubmit.
601	esMD validation error: Procedure Code in response not equal Procedure Code in request. Correct and resubmit.
602	esMD validation error: Approved Service Date must be greater than or equal to current system date.
603	esMD validation error: Decision Indicator = R; response is missing at least one combination of Error Category Code: Error Code. Add the combination(s) of Error Category Code: Error Code and resubmit.
604	esMD validation error: More than 9 Error Codes were reported for a single Error Category Code. Reduce the number of errors for each Error Category Code to 9 and resubmit.

Error Code	Error Description
605	esMD validation error: Decision Indicator = R; Category Code is invalid for the combination of Error Category Code: Error Code. Correct the Error Category Code and resubmit with correct combination(s) of Error Category Code: Error Code.
606	esMD validation error: Decision Indicator = R; invalid Error Code for the combination of Error Category Code: Error Code. Correct the Error Code and resubmit with correct combination(s) of Error Category Code: Error Code.
607	esMD validation error: Invalid Industry Code. Correct and resubmit.
608	esMD validation error: Invalid Reason Code. Correct and resubmit.
609	McAfee Gateway is unavailable. Retry later.
610	esMD validation error: Empty File Received in the Response. Correct and Resubmit.
611	esMD validation error: Multiple Files Received in the Response.
612	esMD validation error: Approved Service Date and Approved Service Date Range and Approved Unit should not be provided for this response. Correct and Resubmit.
613	esMD validation error: Administrative Error Code is invalid. Correct and Resubmit.
614	esMD validation error: Approved Service End Date is less than the Current Date. Correct and resubmit.
615	esMD validation error: Invalid error in the pickup notification.
616	esMD validation error: Intended Recipient OID is deactivated and cannot accept response. Correct and resubmit.
617	esMD validation error: Mailbox ID in the response does not match with the Mailbox ID that the request was sent. Correct and resubmit.
618	ESMD validation error: Error occurred while storing the Review Contractor Status PickUp.
619	ESMD validation error: Error occurred while validating the Review Contractor PickUp Status Data.
620	esMD validation error: Invalid Review Response Creation Time format.
621	esMD validation error: Invalid review Response Submission Time Format.
622	esMD validation error: This Decision Indicator is not valid for this response.
623	esMD validation error: Both Approved Service Date and Approved Service Date range cannot exist in same response. Correct and Resubmit.
624	esMD validation error: Approved Service Start Date cannot be greater than the Approved Service End Date. Correct and Resubmit.
625	esMD validation error: Reason code is not required for decision Indicator A. Correct and Resubmit.
626	esMD validation error: Unable to parse RC response.
627	esMD validation error: Error encountered while saving ReviewContractorPickUpStatus data.
628	esMD validation error: Failure in sending the Administrative error response to HIH.
629	esMD validation error: Failure in sending the Administrative PA response to HIH.
630	esMD validation error: Failure in sending the Pickup notification to HIH.
640	esMD validation error: Intended recipient OID and Procedure Code is not a valid combination. Correct and resubmit.

16.2 Errors: RC to esMD

There are two types of Error Codes sent by the RC to the esMD. They are:

1. Administrative Errors; and
2. Pickup Errors.

16.2.1 Administrative Errors

Table 42: Administrative Error Codes lists the error codes used to report unexpected errors related to the payload received in a downloaded file from esMD. For more details, please refer to section 11.2.5 Administrative Error Response to Inbound Submissions.

Table 42: Administrative Error Codes

Administrative Error	Error Code	Description
corrupt files/cannot read files	ESMD_410	ESMD_410- Administrative Error (corrupt files/cannot read files).
Submission Sent to Incorrect RC	ESMD_411	ESMD_411- Administrative Error (Submission Sent to Incorrect RC).
Virus Found	ESMD_412	ESMD_412- Administrative Error (Virus Found).
Other	ESMD_413	ESMD_413- Administrative Error (Other).

16.2.2 Pickup Errors

Table 43: Pickup Error Codes lists the types of error codes and their descriptions. These codes are used to populate the ErrorInfo object inside the error pickup notification XML (e.g., R_TID_Pickup_Error_Request.xml). Please refer to section 11.2.2 Error Pickup Notification for more details.

Table 43: Pickup Error Codes

Error Type	Error Code	Description
UNZIP ERROR	534	ESMD_534 – RC Client processing error (Unzip failure). Please resubmit.
CHECKSUM ERROR	535	ESMD_535 – RC Client processing error (Checksum issue). Please resubmit.
METADATA ERROR	536	ESMD_536 – RC Client processing error (Metadata issue). Please resubmit.

17. Contacts

Table 44: Support Points of Contact list for esMD.

Table 44: Support Points of Contact

Contact	Phone	Email	Hours of Operation
CMS esMD Help Desk	(443) 832-1856	esMD_Support@cms.hhs.gov	Regular Business Hours: 8 a.m. to 8 p.m. Eastern Time (ET).

Appendix A: Description of Fields on RC Client Tabs

Table 45: Descriptions of Fields on Review Decision Response to PA Request Tab lists the descriptions of the fields on the Review Decision Response to PA Request tab.

Table 45: Descriptions of Fields on Review Decision Response to PA Request Tab

Name of Field	Description
Transaction ID	Transaction Identifier of the request this response is being sent for. Required Element. Minimum Length should be 7 and Maximum Length should be 7. Format of this element is Numeric.
Procedure Code	Procedure Code for the Review Response. It should be the same as the one that was sent in the request. Required Element. Length should be 1 – 48 Alpha – Numeric.
Procedure Level Decision	Decision provided for the Review Response; can be one of the following: • "A" for A - Affirmed; • "M" for M - Affirmed with Change; or • "N" for N - Non Affirmed.
Number of Approved Units	Number of Approved Units for the request service; entered only when Procedure Level Decision is M - Affirmed with Change decision. Maximum Length should be 10. It should be Numeric.
Approved Service Date	Approved Service Date is the new date the RC is approving for the requested service if the response is for M - Affirmed with Change Decision; the format is mm-dd-yyyy and the date must be same or later than the current date; can be selected from the calendar; entered only when Procedure Level Decision is M - Affirmed with Change decision.
Approved Service Date Range	Approved Service Date Range for approving the requested service if the response is for M - Affirmed with Change Decision; the format is mm-dd-yyyy; can be selected from the calendar with Start Date and End date; entered only when Procedure Level Decision is M - Affirmed with Change decision.
• **Start Date**	Start Date is the approved service start date for approving the requested service if the response is for M - Affirmed with Change Decision; the format is mm-dd-yyyy; can be selected from the calendar; Can be Past, current or Future date; and is entered only when Procedure Level Decision is M - Affirmed with Change decision.
• **End Date**	End Date is the approved service end date for approving the requested service if the response is for M - Affirmed with Change Decision; the format is mm-dd-yyyy; can be selected from the calendar; Should be greater than the Start Date provided and should be same or later than current date; and is entered only when Procedure Level Decision is M - Affirmed with Change decision.
Procedure Level UTN	Unique Tracking Number for review response; Required Element for all decisions; format of the unique tracking number is 1 – 50 Alpha Numeric.
Industry Code	Industry Code(s) if the decision is Non Affirmed for the response; minimum one and maximum of five industry code(s).

Name of Field	Description
Reason Code	Reason code; required only when Procedure Level Decision is N - Non Affirmed; optional when Procedure Level Decision is M - Affirmed With Change; format is five character alpha number; and a minimum of 1 and up to maximum of 25 reason codes can be provided.

Table 46: Descriptions of Fields on Error Response to PA Request Tab lists the descriptions of the fields on the Review Decision Response to PA Request tab.

Table 46: Descriptions of Fields on Error Response to PA Request Tab

Name of Field	Description
Transaction ID	Transaction Identifier of the request that is associated with the response that is being sent to the request. Required Element. Minimum Length should be 7 and Maximum Length should be 7. Format of this element is Numeric.
Reject Error Category	One or multiple Reject Error Category is selected for each Response; each Reject Error Category has number of Reject Error Codes associated with it. Required Element.
Reject Error Code	Under Each Reject Error Category, either one or multiple Reject Error Codes are selected. Required Element. Minimum 1 and maximum 9 reject error codes can be selected for each category.
Reason Code	5-character reason code is provided. Minimum of 1 and up to maximum of 25 reason codes can be provided. Required Element.
Request Level UTN	UTN is provided for each response. Optional Element. Format of the unique tracking number is 1 – 50 Alpha Numeric Characters.

Table 47: Descriptions of Fields on Administrative Error Response to Inbound Submissions Tab lists the descriptions of the fields on the Administrative Error Response to Inbound Submissions tab.

Table 47: Descriptions of Fields on Administrative Error Response to Inbound Submissions Tab

Name of Field	Description
Transaction ID	Transaction Identifier of the request this response is being sent for. Required Element. Minimum Length should be 7 and Maximum Length should be 7. Format of this element is Numeric
Error Situation	Error code/situation; can be one of the following: • Corrupt Files/Cannot read file, • virus found, • Submission sent to incorrect RC, or • Other. Required Element.

Table 48: Descriptions of Fields on Advanced/Debugging Tab lists the descriptions of the fields on the Advanced/Debugging tab.

Table 48: Descriptions of Fields on Advanced/Debugging Tab

Name of Field	Description
User ID	EIDM User ID Required Element for testing the connectivity to TIBCO.
Password	EIDM password Required Element for testing the connectivity to TIBCO.

Appendix B: Reject Error Codes

For an up-to-date list of Reject Error Codes, please refer to the esMD Downloads section, using the link below:

http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Information_for_Review-Contractors.html

(*Note: An up-to-date list of Reject Error Codes will be added to this web site by CMS.*)

Appendix C: Industry Codes

For an up-to-date list of Industry Codes, please refer to the esMD Downloads section, using the link below:

http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Information_for_Review-Contractors.html

(*Note: An up-to-date list of Industry Codes will be added to this web site by CMS.*)

Appendix D: PA Reason Codes

For an up-to-date list of PA Reason Codes, please refer to the esMD Downloads section, using the link below:

http://www.cms.gov/Research-Statistics-Data-and-Systems/Computer-Data-and-Systems/ESMD/Information_for_Review-Contractors.html

(*Note: An up-to-date list of PA Reason Codes will be added to this web site by CMS.*)

Appendix E: PA Procedure Codes

Table 49: Procedure Codes for the PA Programs lists the Procedure Codes for the PA programs for the X12 requests and responses.

Table 49: Procedure Codes for the PA Programs

Procedure Code	PA Program
K0800	Power Mobility Device
K0898	Power Mobility Device
K0891	Power Mobility Device
K0890	Power Mobility Device
K0855	Power Mobility Device
K0854	Power Mobility Device
K0853	Power Mobility Device
K0852	Power Mobility Device
K0851	Power Mobility Device
K0850	Power Mobility Device
K0849	Power Mobility Device
K0848	Power Mobility Device
K0843	Power Mobility Device
K0842	Power Mobility Device
K0841	Power Mobility Device
K0840	Power Mobility Device
K0839	Power Mobility Device
K0838	Power Mobility Device
K0837	Power Mobility Device
K0836	Power Mobility Device
K0835	Power Mobility Device
K0802	Power Mobility Device
K0812	Power Mobility Device
K0813	Power Mobility Device
K0814	Power Mobility Device
K0815	Power Mobility Device
K0816	Power Mobility Device
K0817	Power Mobility Device
K0818	Power Mobility Device
K0819	Power Mobility Device
K0820	Power Mobility Device
K0821	Power Mobility Device
K0822	Power Mobility Device
K0823	Power Mobility Device
K0824	Power Mobility Device
K0825	Power Mobility Device
K0826	Power Mobility Device

Procedure Code	PA Program
K0827	Power Mobility Device
K0828	Power Mobility Device
K0829	Power Mobility Device
G0277	Hyperbaric Oxygen (HBO) Therapy
A0426	Non-Emergent Ambulance Transport
A0428	Non-Emergent Ambulance Transport
K0856	Durable Medical Equipment Prosthetics Orthotics and Supplies
K0861	Durable Medical Equipment Prosthetics Orthotics and Supplies
G0161	Home Health Services
G0153	Home Health Services
G0160	Home Health Services
G0158	Home Health Services
G0152	Home Health Services
G0159	Home Health Services
G0157	Home Health Services
G0151	Home Health Services
G0156	Home Health Services
G0299	Home Health Services
G0300	Home Health Services
G0162	Home Health Services
G0163	Home Health Services
G0155	Home Health Services
G0164	Home Health Services

Appendix F: Data Directories

Table 50: Inbound File Names and Data Directories lists out all the files received by the RC and the corresponding data directories these files will reside in along with a brief description.

Note: "TID" indicates a "Transaction ID" in the table below.

Table 50: Inbound File Names and Data Directories

Data Directory	Folder	File Name	Description
acknowledgement	A_TID	A_TID_Pickup_HIH_Status_Response.xml	Successful Acknowledgement that the HIH has received the pickup notification.
error	M_TID	M_TID_Administrative_Response_Validation_Error.xml	Validation Error received by the RC for the Administrative Error Response sent to esMD.
error	R_TID	R_TID_Pickup_Validation_Error.xml	Validation Error received by the RC for the Pickup Notification sent to esMD.
error	V_TID	V_TID_PA_Review_Response_Validation_Error.xml	Validation Error received by the RC for the PA Review Response sent to esMD.
error	X_TID	X_TID_Virus_Scan_Error.xml	Virus Scan Failure Error received by the RC from the esMD on a file sent previously.
input	E_TID	E_TID-UniqueIdDateTimestamp_sequence.pdf	Documentation received as part of the inbound submission i.e. PDF.
input	E_TID	E_TID-flatfilerendering.ffr	Flat File Rendering of the X12N 278 PA Request received by the RC from esMD.
input	E_TID	E_TID-coversheet.pdf	Coversheet PDF of the X12N 278 PA Request received by the RC from esMD.
Input	E_TID	E_TID-metadata.xml	Metadata XML for the payloads received by the RC in the inbound submission from esMD.
notifications	N_TID	N_TID_PA_Review_Result_HIH_Status_Response.xml	Successful Acknowledgement that the HIH has received the PA Review Response.
notifications	S_TID	S_TID_Administrative_Error_HIH_Status_Response.xml	Successful Acknowledgement that the HIH has received the administrative error response.

Table 51: Outbound File Names and Data Directories lists all the outbound files sent from the RC to the esMD along with a brief description and the data directory where they are created.

Note: "TID" indicates a "Transaction ID" in the table below.

Table 51: Outbound File Names and Data Directories

Data Directory	Folder	File Name	Description
output	P_TID	P_TID_Pickup_Notification.xml	Successful pickup notification being sent to esMD inside the zip file.
output	P_TID	P_TID_Pickup_Error_Notification.xml	Error pickup notification (i.e. unzip, checksum etc.) being sent to esMD inside the zip file.
output	D_TID	D_TID_Administrative_Error_Response.xml	Administrative Error Response being sent to esMD inside the zip file.
output	E_TID	E_TID_PA_Review_Response.xml	PA Review Response being sent to esMD inside the zip file.

Appendix G: Content Type Codes

Table 52: Content Type Code Descriptions provides the description of the Content Type Code used in the esMD Release 4.0.

Table 52: Content Type Code Descriptions

Content Type Code	Description	Comment
1	Response to Additional Documentation Request(ADR)	N/A
8	PMD PA	N/A
9	First Level Appeal Requests	N/A
9.1	Second Level Appeal Requests	N/A
10	ADMC	N/A
11	RA Requests	N/A
12	Supporting Documentation for the unsolicited X12N 278 Request	N/A
13	Supporting Documentation for the X12N 278 Request	N/A
8.1	Non-Emergent Ambulance Transport	Will accept 81 till a period determined by CMS.
8.2	Hyperbaric Oxygen (HBO) Therapy	Will accept 81 till a period determined by CMS.

Table 53: Content Type Codes and Business Types shows the list of Business Types associated with them.

Table 53: Content Type Codes and Business Types

Content Type Code	Business Type
1	Response message for additional documentation request
8	Requesting PMD PA
9	Requesting a First Level Appeal
9.1	Requesting a Second Level Appeal
10	ADMC
11	RA Requests
8.1 or 81	Requesting Non-Emergent Ambulance Transport
8.2 or 82	Requesting Hyperbaric Oxygen (HBO) Therapy
12	Unsolicited supporting documentation using XDR Profile
13	XDR X12

Appendix H: Record of Changes

Table 54: Record of Changes

Version Number	Date	Author/Owner	Description of Change
1.0	04/25/2016	Srini Eadara, Vijayalakshmi Muthukrishnan	Updated for Release AR2016.07.0
1.1	05/09/2016	Vijayalakshmi Muthukrishnan	Addressed CMS comments for the version 1.0

Appendix I: Acronyms

Table 55: Acronyms

Acronym	Literal Translation
ADMC	Advance Determination of Medicare Coverage
ADR	Additional Documentation Request
API	Application Programming Interface
CMS	Centers for Medicare & Medicaid Services
DMEPOS	Durable Medical Equipment, Prosthetics, Orthotics and Supplies
EFT	Enterprise File Transfer
EIDM	Enterprise Identity Management
esMD	Electronic Submittal of Medical Documentation
FFR	Flat File Rendering
FFS	Fee-For-Service
HBO	Hyperbaric Oxygen
HIH	Health Information Handler
HHS	Home Health Services
JAR	Java Archive
JCE	Java Cryptography Encryption
JDK	Java Development Kit
JKS	Java KeyStore
JRE	Java Runtime Environment
MB	Megabytes
MFT	Managed File Transfer
PA	Prior Authorization
PDF	Portable Document Format
PMD	Power Mobility Device
RC	Review Contractor
RSA	Rivest, Shamir & Adleman
SFTP	SSH File Transfer Protocol
SSH	Secure Shell
URL	Universal Resource Locator
UTN	Universal Tracking Number
XDR	Cross-Enterprise Document Reliable Interchange
XML	Extensible Markup Language

Appendix J: Glossary

Table 56: Glossary

Glossary	Description
Additional Documentation Request (ADR)	Official letters sent to Providers from CMS RCs requesting additional documentation that is needed to process claims.
Advanced Determination of Medical Coverage (ADMC)	A voluntary program that allows Suppliers and Beneficiaries to request prior approval of eligible items (e.g., wheelchairs) before delivery of the items to the beneficiary.
CONNECT	CONNECT implements a flexible, open-source gateway solution that enables healthcare entities - Federal agencies or private-sector health organizations or networks - to connect their existing health information systems to the eHealth Exchange. CONNECT is fully functional out-of-the-box, while at the same time configurable and flexible to allow organizations to customize it to meet their needs and those of their existing health information systems.
Electronic Submission of Medical Documentation (esMD)	A new mechanism for submitting medical documentation via a secure internet gateway connecting Providers to the Centers for Medicare & Medicaid Services (CMS). In its second phase, esMD will allow Medicare RCs to electronically submit claim related Additional Document Request (ADR) letters, and other use case requests, to Providers when their claims are selected for review.
Health Information Handler (HIH)	A Health Information Handler (HIH) is defined as an organization that oversees and governs the exchange of health-related claim reviewer information from Provider to CMS esMD Gateway according to nationally recognized standards.
Interface	A well-defined boundary where direct contact between two different environments, systems, etc., occurs, and where information is exchanged.
Power Mobility Device (PMD) Prior Authorization (PA)	The CMS implemented a Prior Authorization process for scooters and power wheelchairs for people with Fee-For-Service Medicare who reside in seven states with high populations of fraud- and error-prone Providers (CA, FL, IL, MI, NY, NC, and TX). This demonstration will help ensure that a beneficiary's medical condition warrants their medical equipment under existing coverage guidelines. Moreover, the program will assist in preserving a Medicare beneficiary's ability to receive quality products from accredited suppliers.
Security	The physical, technological, and administrative safeguards used to protect individually identifiable health information.
SOAP	Simple Object Access Protocol is a message exchange format for web services.
Transaction	Event or process (such as an input message) initiated or invoked by a user or system, regarded as a single unit of work and requiring a record to be generated for processing in a database.

Appendix K: Referenced Documents

Table 57: Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
None.		

Appendix L: Approvals

The undersigned acknowledge that they have reviewed the Review Contractor (RC) Client Java User Guide and Installation Handbook, Version 1.1 Final, and agree with the information presented within this document. Changes to this Guide will be coordinated with, and approved by, the undersigned, or their designated representatives.

Signature:		Date:	05/12/2016
Print Name:	Maureen Hoppa		
Title:	Contracting Officer's Representative		
Role:	CMS Approving Authority		