**Asurion Subscriber Billing System SQA Testing Specification**

**Versio****n: 2.0**

**Revision History**

**Change Record**

| Date | Author | Version | Changes |
| --- | --- | --- | --- |
| 12/01/2011 | Jim Emerson | 0.1 | Draft version |
| 12/23/2011 | Jim Emerson | 1.0 | Updated |
| 3/12/2012 | Jim Emerson | 2.0 | Included all requested changes. |
|  |  |  |  |
|  |  |  |  |

**Document Properties**

| Item | Details |
| --- | --- |
| Document Title | Asurion Subscriber Billing System SQA Testing Specification |
| Author | Jim Emerson |
| Creation Date | 12/01/2011 |

**Table of Contents**

[Introduction 5](#_Toc312400773)

[Asurion Subscriber Billing System – Finance Services 5](#_Toc312400774)

[Monitoring and Logging 6](#_Toc312400775)

[Monitoring with Administrator 6](#_Toc312400776)

[Monitoring Management 6](#_Toc312400777)

[Monitoring with Hawk 7](#_Toc312400778)

[Monitoring Rulebases 10](#_Toc312400779)

[Monitoring Connection Counts 11](#_Toc312400780)

[Monitoring BW Engine Exceptions 11](#_Toc312400781)

[Monitoring BW Application Instance Restarts 12](#_Toc312400782)

[Monitoring Asurion Application Error DB Inserts 12](#_Toc312400783)

[Collecting BW Historical Data 13](#_Toc312400784)

[Viewing a BW Log File Not Rolled From the Past 7-14 Days 13](#_Toc312400785)

[Monitoring with PRTG 13](#_Toc312400786)

[PRTG Network Monitoring 13](#_Toc312400787)

[Logging Services 15](#_Toc312400788)

[Incident Management and Error Reporting 15](#_Toc312400789)

[Incident Management 15](#_Toc312400790)

[Problem Issue and Error Reporting 15](#_Toc312400791)

[CIRT Reports 16](#_Toc312400792)

[Alerts 17](#_Toc312400793)

[BW Hawk Alerts 17](#_Toc312400794)

[Administrator Alerts 18](#_Toc312400795)

[SQA and Testing 19](#_Toc312400796)

[Billing Manager Tests 19](#_Toc312400797)

[Test: Exercise Subscriber Billing System 20](#_Toc312400798)

[Sample Test 21](#_Toc312400799)

[Smoke Tests 22](#_Toc312400800)

[Test: Creating an Enrollment 22](#_Toc312400801)

[Regression Tests 23](#_Toc312400802)

[Test: Failure to Pay Triggers Automatic Cancel Contract Letter 23](#_Toc312400803)

[End to End and Integration Tests 24](#_Toc312400804)

[Test: ProcessNonRefPayment Service 24](#_Toc312400805)

[Appendix 25](#_Toc312400806)

[A.1 Asurion Subscriber Billing System Error Messages 25](#_Toc312400807)

[A1.1 .NET Error Messages and Meanings 25](#_Toc312400808)

[A1.2 CRM Error Messages and Meanings 25](#_Toc312400809)

[CreateContract API Error Messages 26](#_Toc312400810)

[Index 27](#_Toc312400811)

**Figures**

[Figure 1. Asurion Finance Service Gateway 5](#_Toc312400821)

[Figure 2. Administrator - Monitoring Management 6](#_Toc312400822)

[Figure 3. Administration Tool - CreateContract Service Instance 7](#_Toc312400823)

[Figure 4. Hawk - Monitoring Agents 7](#_Toc312400824)

[Figure 5. Hawk Monitoring Agent Display 8](#_Toc312400825)

[Figure 6. Hawk MicroAgent Display 9](#_Toc312400826)

[Figure 7. Hawk - Method Level Detail 9](#_Toc312400827)

[Figure 8. Hawk - Rulebase Monitoring 10](#_Toc312400828)

[Figure 9 PRTG Network Monitor 14](#_Toc312400829)

[Figure 10 Hawk IO Exception 14](#_Toc312400830)

[Figure 11. Daily CIRT Report (1) 16](#_Toc312400831)

[Figure 12. Daily CIRT Report (2) 16](#_Toc312400832)

[Figure 13. NOC CIRT P6 Incidents Summary Report 17](#_Toc312400833)

[Figure 14. BW Hawk Alerts 17](#_Toc312400834)

[Figure 15. Administrator - Monitoring Management 18](#_Toc312400835)

[Figure 16. Monitoring Management Selection 18](#_Toc312400836)

[Figure 17. Sample: Administrator Alert 19](#_Toc312400837)

[Figure 18 Walmart Test 22](#_Toc312400838)

# Introduction

This section introduces the Asurion Subscriber Billing System Software Quality Assurance (SQA) Testing Service and procedures. The following sections provide an overview of the Asurion Subscriber Billing System, describe monitoring and logging processes used with the system, and explain SQA applications and procedures used to test the Subscriber Billing System.

## Asurion Subscriber Billing System – Finance Services

The Asurion Subscriber Billing System incorporates automatic and manually controlled finance service operations. Automatic “CRM” processes flow after customer enrollment through the Asurion Finance Service Gateway. Authorized users control manual “NET” processes via Tibco BusinessWorks (BW) screens. The Subscriber Billing System has a Finance Services Gateway with a Finance Service Agent and methods that interact with a CreateContractGateway, ProcessNonRefPaymentGateway, and CalculateTaxesGateway. Asurion uses SQA Test Service procedures to test the functionality and performance of these services.

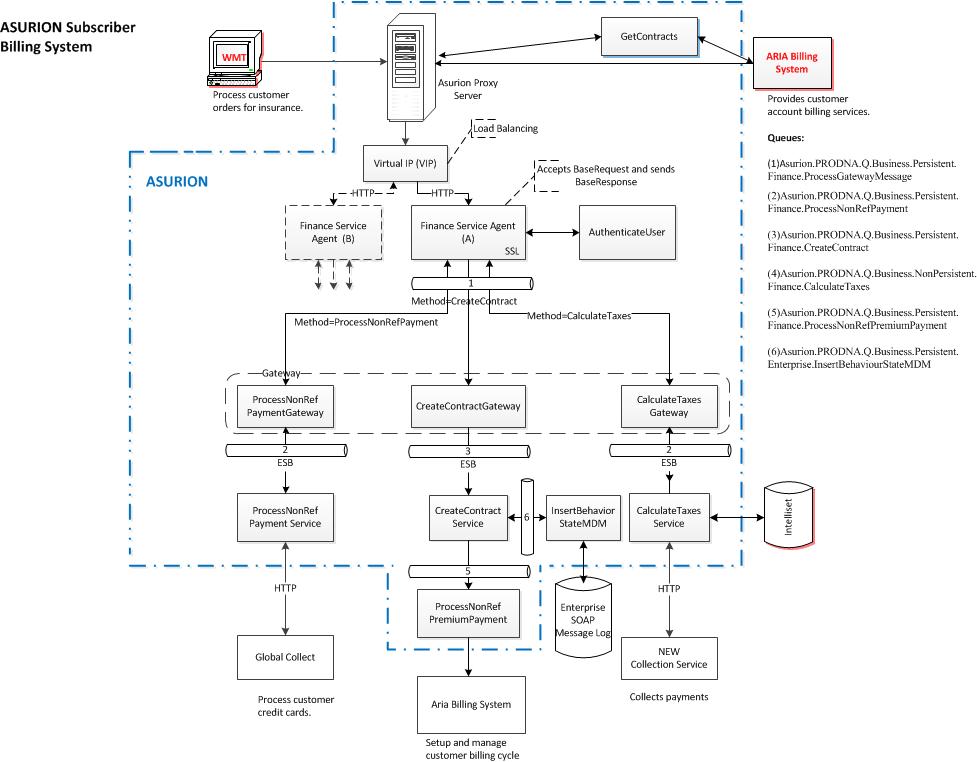


Figure . Asurion Finance Service Gateway

For more information about these process flows and operations, see the “Asurion Subscriber Billing System Product Specification” and “Asurion Subscriber Billing System API Specification.”

# Monitoring and Logging

Asurion uses TIBCO BusinessWorks (BW) HAWK to monitor the health and performance of the Asurion Subscribevr Billing System operations. Treating the entire network as one system, HAWK provides a broad-ranging approach to monitoring the TIBCO Enterprise Messaging Service (EMS) and EMS server operations involved in Finance Service Gateway activities.

The Asurion NOC uses Paessler Router Traffic Grapher (PRTG) to monitor Asurion networks, bandwidth usage, and applications. These monitoring and logging processes are described in the following sections.

## Monitoring with Administrator

Asurion uses Tibco tools to monitor many Subscriber Billing System operations. The Tibco Administrator is used to manually monitor FinanceService applications running on BW servers including:

* CalculateTaxes
* CalculatgeTaxesGateway
* CreateContract-Walmart
* CreateContractGateway
* CreateContractNote
* FinanceServiceAgent
* GetBusinessDays
* GetPayments
* ProcessDunning
* ProcessNonRefPayment
* ProcessNonRefPaymentGateway
* ProcessNonRefPremiumPayment
* ProcessPreDunning
* ProcessPremium
* ProcessRefPayment

The following figures represent screens that the Asurion SQA team uses to access these operations.

### Monitoring Management

Asurion SQA team members use the Administrator Monitoring Management screen to select areas for monitoring.



Figure . Administrator - Monitoring Management

For example, selecting CreateContract-Walmart on the Application Management – All Applications screen leads to the following Service Instances screen:



Figure . Administration Tool - CreateContract Service Instance

Here, the monitor indicates that the application is running with an “OK” Status on a certain machine under BW software.

## Monitoring with Hawk

The Asurion team uses the BW Hawk Console to monitor Agents, Microagents, and Rulebases for Finance Service applications.

When Hawk is used to monitor applications services at the Agent level, information is displayed at the highest level (domain view). For example,

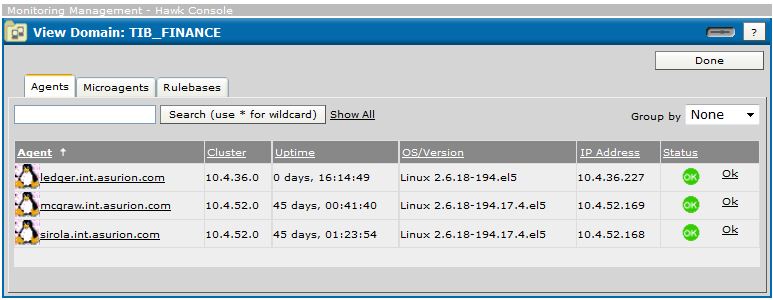


Figure . Hawk - Monitoring Agents

This screen lists particular agents, identifying their server cluster, uptime, operating system version, IP address and current status. Clicking on a particular Agent leads to a list of Microagent.

For example:

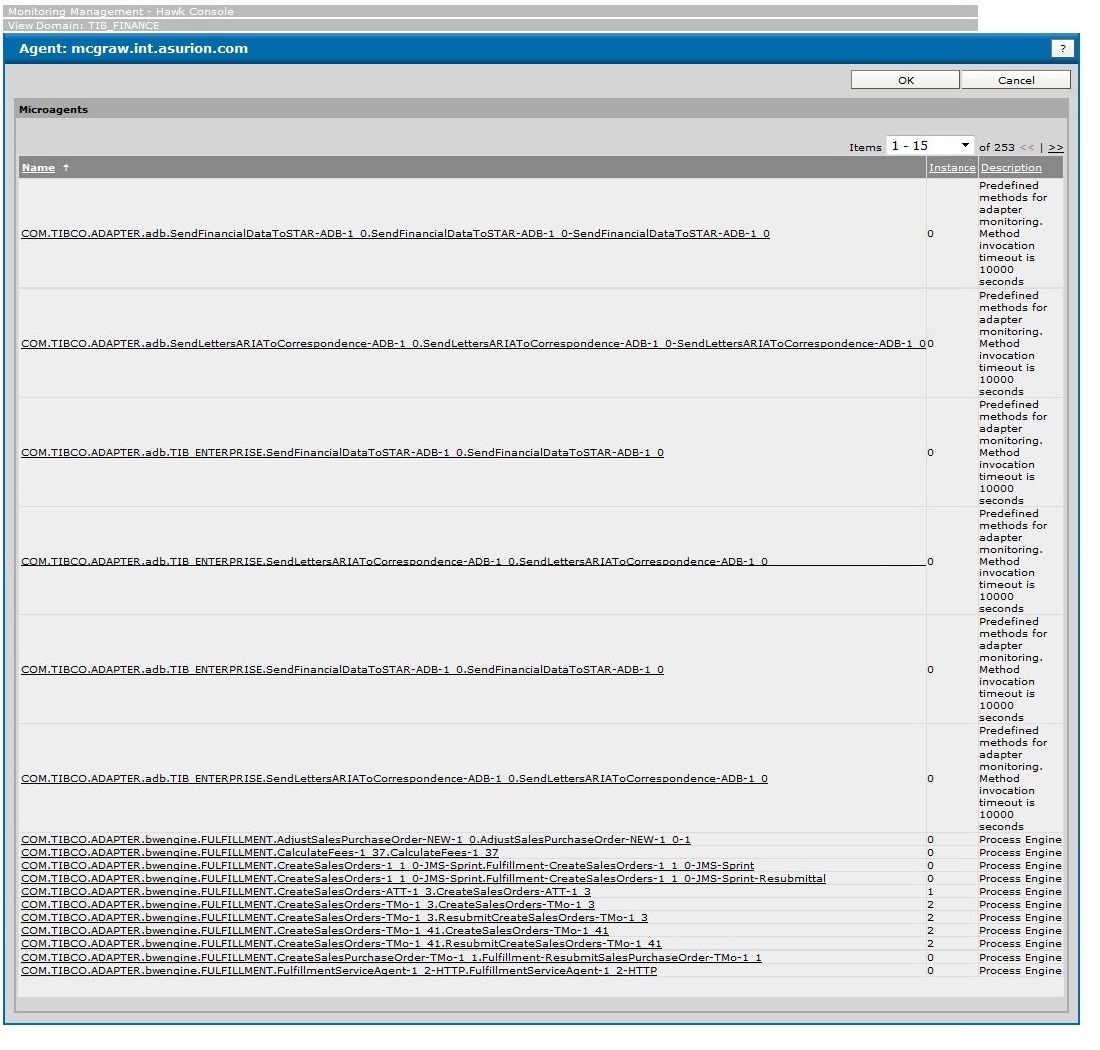


Figure . Hawk Monitoring Agent Display

On this screen, the user can select a specific microagent for more details.

For example,

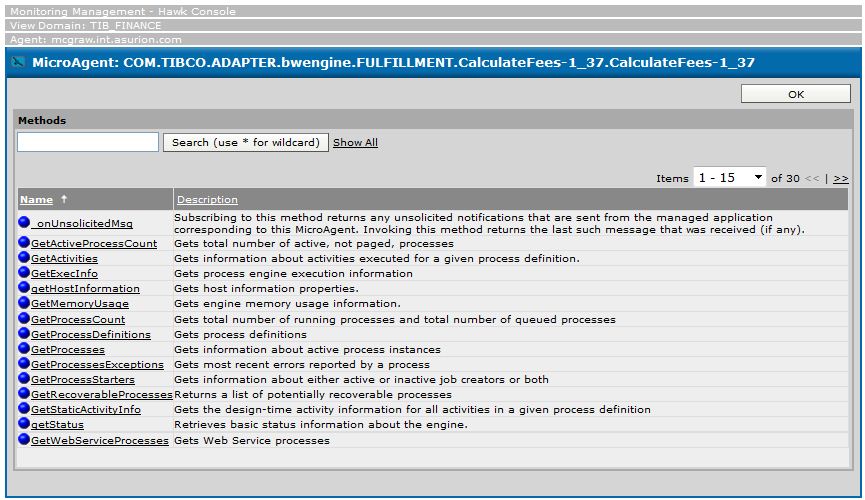


Figure . Hawk MicroAgent Display

At this level, Hawk lists currently available service methods. The user can select a method for detailed information. For example,

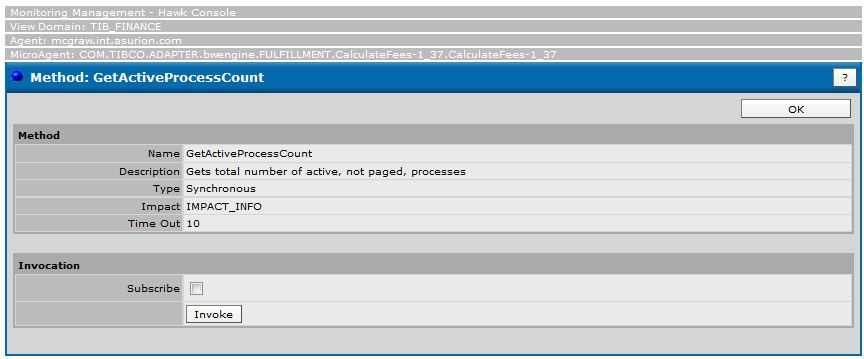


Figure . Hawk - Method Level Detail

**Also See**

“BW Hawk Alerts” in this document.

### Monitoring Rulebases

Asurion also uses BW Hawk to monitor Gateway activity according to the following Rulebases, previously implemented using the Hawk Rulebase Editor:

* Connection counts
* BW engine exceptions
* BW application instance restarts
* Asurion application error DB inserts
* BW historical data collection
* BW log files not rolled from the past 7-14 days

The Asurion team uses the Hawk Rule Editor to define the monitoring rules (for example, methods to use) for the Rulebases. Hawk accesses information stored in log files.

**Rulebase Monitoring Example**

When the user selects the Hawk Rulebases tab, the screen lists the currently used rulebases. Clicking the Rulebases tab on the Administrator leads to a screen listing all currently running rulebases. Then, selecting a single rulebase displays a screen with details about the rulebase.

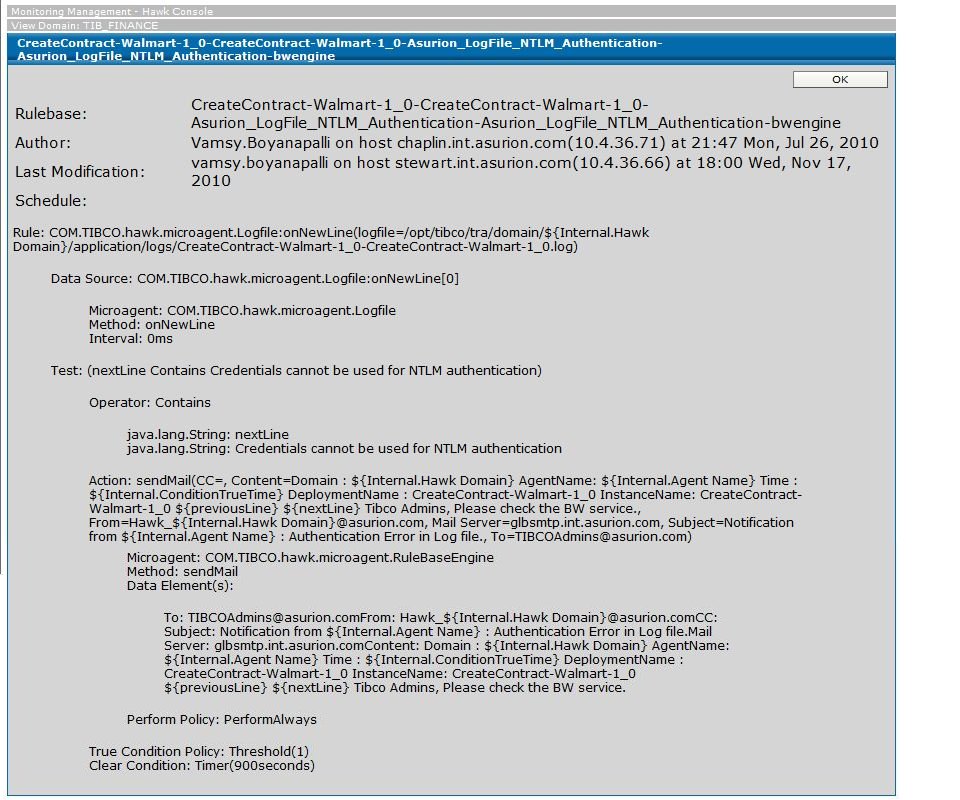


Figure . Hawk - Rulebase Monitoring

Here, Hawk displays the following information for a CreateContract Authentication BW engine rulebase: rulebase name, author, last modification date, schedule, and rule information (data source and test items). See “Monitoring BW Engine Exceptions”, below.

### Monitoring Connection Counts

Asurion applies a predefined Hawk rule (Asurion\_TIBCO\_BWAppLog\_Exception-Asurion\_TIBCO\_BWAppLog\_Exception-bwengine) to monitor BW log files and determine connection counts and associated exceptions.

* The rule determines if connection count on a given EMS instance is greater than 3500. The monitoring process sends an email alert message to members of a distribution list and an alert to the Hawk console (Tibco Administrator).
* The rule also monitors BW log files for the following exceptions:
* “A Resource Required By Service Is Not Available”
* “Activity timed out”
* “ActivityTimedOutException”
* “SOAP Fault Response”
* “Input data invalid”
* “IOException”
* “Cannot authenticate user”
* “Client sent invalid username/password”
* “CSIApplicationException”
* “Faultcode”
* “No TMo SKU was found for the given SKU”
* “No data was found for the search criteria”
* “Validation error”
* “Login failed”
* “Failed to create secure client socket”
* “Deserializing/processing the incoming SOAP message”
* “MESSAGE\_PROCESSOR\_INSTANTIATION\_ERROR”
* “Server memory limit exceeded exception”
* When an exception is found in the log files, the rule also prompts a return message with explanatory details about the exception.

### Monitoring BW Engine Exceptions

Following a predefined rule (*BW Instance*-OutOfMemory\_ JMSConnTerminated\_ ErrorState\_Restart-bwengine), Hawk is used to monitor BW engine exceptions. The rule monitors:

* Out of Memory Exception—Checks the BW instance log for instances of “java.lang.OutOfMemoryError: Java heap space”. If it detects errors, posts the condition as “True.”
* JMS Connection Terminated—Checks the BW instance log for instances of “Connection has been terminated”. If so, runs a shell script to end that BW instance and calls the method “Insert\_BW\_Errors” to insert error details into the database.
* Error State—Determines if the BW instance is in an error state. If so, runs a shell script to end that BW instance and calls the method “Insert\_BW\_Errors” to insert error details into the database.

### Monitoring BW Application Instance Restarts

Hawk follows a predefined rule (*BW Instance*-OutOfMemory\_ JMSConnTerminated\_ ErrorState\_Restart-Asurion\_BW\_ADB\_Restart-bwengin) to monitor BW instance stops (state STOPPED) and restarts. The rule:

* Stops and Restarts—Determines when the BW application is in the STOPPED state. When this state occurs, Hawk issues an alert message, runs a shell script to end that BW instance, and calls the method “Insert\_BW\_Errors” to insert error details into the database.

### Monitoring Asurion Application Error DB Inserts

Hawk follows a predefined rule (Asurion\_AppErrors\_DBInsert-Asurion\_AppErrors\_DBInsert-bwengine) to monitor the insertion of Error or Exception strings and associated details into the database. The rule sets the monitor to detect the following errors and exceptions:

* A Resource Required By Service Is Not Available
* Activity timeout
* Activity Timed Out Exception
* SOAP Fault Response
* Input data invalid
* IOException
* Cannot authenticate user
* Client invalid username/password
* CSI Application Exception
* Fault code
* No TMo SKU was found for the given SKU
* No data was found for the search criteria
* Validation error
* Login failed
* Failed to create secure client socket
* Deserializing/processing the incoming SOAP message
* MESSAGE\_PROCESSOR\_INSTANTIATION\_ERROR
* Server memory limit exceeded exception
* Could not create connection with the JMS server
* Failed to connect to any server
* ClientID already exists
* Service not available
* Termination
* Out of Memory Exception
* JMS Connection Terminated
* Javax.jms.IllegalStateException:Connectionisclosed)

When one of these error/exceptions occurs, Hawk issues an alert message, runs a shell script to end that BW instance, and calls the method “Insert\_BW\_Errors” to insert error details into the database.

### Collecting BW Historical Data

Hawk follows a predefined rule (BWHISTORICAL-DATA-PROD-BW-Historical-Data-bwengine) to insert historical BW activity information into the database. The four-part rule:

1. Checks for a successful ping to the DB, and then posts the condition “Exists”.
2. If the condition is “Exists”, calls the method “insert\_BW\_MemoryUsage” to insert historical activity data into the DB.
3. If the ping to the DB is unsuccessful, the rule sends an alert email to the DB team.

### Viewing a BW Log File Not Rolled From the Past 7-14 Days

Hawk follows a predefined rule (Asurion\_TIBCO\_BWAppLog\_NotRolled-Asurion\_TIBCO\_BWAppLog\_NotRolled-bwengine) to monitor the log file of BW applications.

* When a log file has not been closed and a new file opened (log file rolling) for the past 7 to 13 days, the rule sends an informational alert email to the Tibco administration team.
* If the log file has not been rolled for 14 or more days, the rule sends an escalated alert email. The MARS utility is used to implement the rule.

**Also See**

“Tibco BW\_Hawk\_Rules\_Description.doc” at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6DA69BD2-25DE-48A5-A0B8-039239D5C7F8}](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6DA69BD2-25DE-48A5-A0B8-039239D5C7F8%7d).

## Monitoring with PRTG

NOC uses the Paessler Router Traffic Grapher (PRTG) to monitor the condition of Asurion networks, bandwidth usage, and applications.

### PRTG Network Monitoring

PRTG monitors Asurion system alerts recorded by Tibco Hawk in a database. Responding to alerts, PRTG sends notifications and data to the NOC central monitoring dashboard.

PRTG monitors reports from the following Tibco Hawk sensors every 5 minutes:

* IO Exception
* CSI Application Exception
* SOAPFaultResponse
* Inputdatainvalid
* AResourceRequiredByServiceIsNotAvailable
* Activitytimeout

When a sensor exceeds a defined threshold, PRTG sends an email alert to the Tibco support team and a notification to the NOC Spectrum Dashboard. The dashboard automatically creates a service desk ticket. NOC may promote a ticket to a Critical Incident Response Team (CIRT) for resolution. In the Mission Control room, an on-duty Incident Controller monitors the PRTG central dashboard for possible problems.

The following figure shows a sample full view of the PRTG Network Monitor.

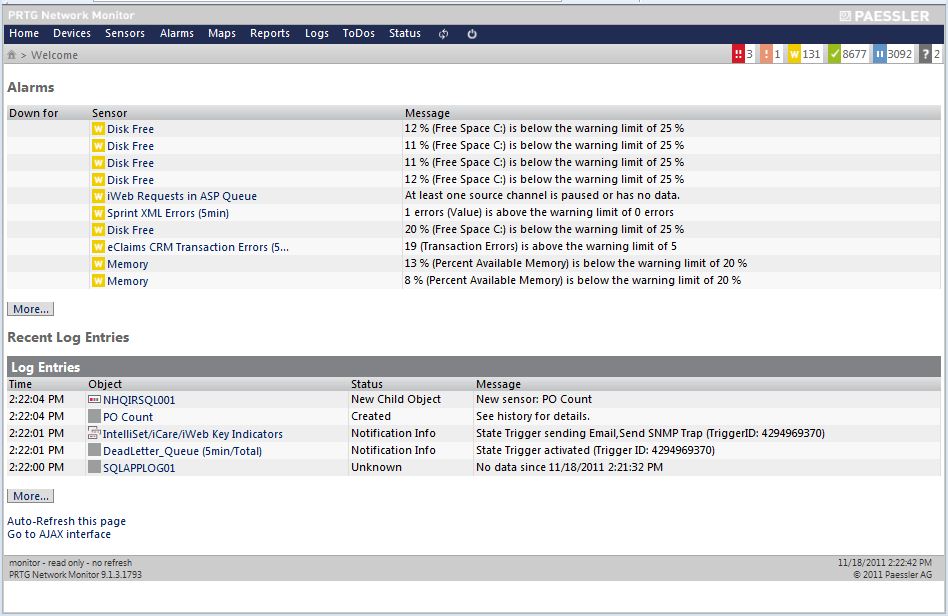


Figure PRTG Network Monitor

Each Hawk sensor provides shows historical, directions on escalating alerts, instructions on problem research, and sometimes, notes on previous incidents and resolutions. The following sample shows a 5 minute Hawk IO Exception report.

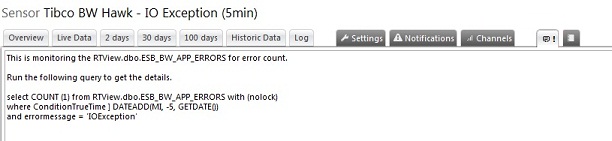


Figure 10 Hawk IO Exception

**Also See**

“PRTG Alert Framework” at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6DA69BD2-25DE-48A5-A0B8-039239D5C7F8}](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6DA69BD2-25DE-48A5-A0B8-039239D5C7F8%7d) and <http://prtg/m/home.htm> for a view of the PRTG Network Monitor.

## Logging Services

The results of monitoring are recorded in the domain folder (for example, PROD\_FINANCE) in the host server. These Hawk logs are maintained and updated every second and minute. Asurion uses the Tibco `Administrator to monitor, manage, deploy, and undeploy monitoring archives (MARs). The team also uses the Tibco Enterprise RTView dashboard console to display the current status of various BW and EMS servers.

# Incident Management and Error Reporting

Asurion product development, production deployment, and maintenance support involves reliable systems of error reporting, SQA management, and functional testing. This section summarizes these closely related services.

## Incident Management

Incident Management (IM) processes enables Asurion SQA to identify an issue or problem, assign responsibility for researching and resolving the issue, and send out appropriate notifications. A typical IM ticket can flow involve the following steps:

* Receive incident notice (from emails, monitoring systems, and phone calls)
* Create an incident ticket
* Collect ticket details
* Determine if the ticket will be further processed or cancelled
* Categorize and prioritize the ticket (P1, P6, and so on)
* Determine if a CIRT will be created
* Determine if the issue is extreme or not
* Decide to involve NOC or not
* Assign to appropriate team for issue troubleshooting and resolution

**Also See**

“IncidentManagement\_Sample.vsd,” “IM V2.0” tab, at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6da69bd2-25de-48a5-a0b8-039239d5c7f8}&RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&TreeField=Folders&TreeValue=Asurion%20Subscriber%20Billing%20System%20Specs&ProcessQStringToCAML=1&SortField=Modified&SortDir=Desc](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6da69bd2-25de-48a5-a0b8-039239d5c7f8%7d&RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&TreeField=Folders&TreeValue=Asurion%20Subscriber%20Billing%20System%20Specs&ProcessQStringToCAML=1&SortField=Modified&SortDir=Desc)

## Problem Issue and Error Reporting

Asurion uses Daily CIRT and Alert reports to identify errors in Subscriber Billing System development, implementation, and operations.

### CIRT Reports

Twice a day (currently, at 12am and 12pm EST), the system automatically generates a Critical Incident Response Team (CIRT) report. Each incident cited in the report is summarized in terms of a CIRT number, incident controller, brief description, resolution/latest updates, current status, start time, and end time (or none, if the system is still running). The following figure shows a sample daily CIRT report.

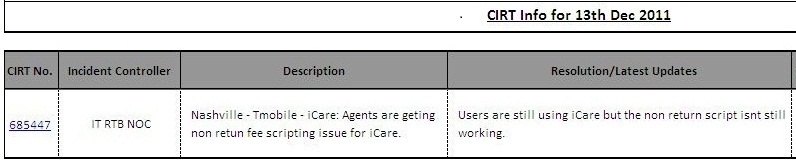


Figure . Daily CIRT Report (1)

and



Figure . Daily CIRT Report (2)

The Asurion NOC team issues reports, summarizing all P6-level incidents from CIRTS in the past day, identifying the responsible team and number of incidents.

For example:

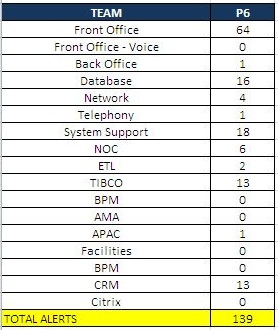


Figure . NOC CIRT P6 Incidents Summary Report

**Also See**

Instructions on creating a CIRT email at the Syswiki site <https://syswiki.int.asurion.com/index.php/CIRT_-_Email_Creation>.

### Alerts

The Asurion team relies upon alerts from the BW Hawk and Tibco Administrator systems.

#### BW Hawk Alerts

BW Hawk for the Subscriber Billing System collects data on operations and writes it to a database. PRTG monitors these alert messages and sends the resulting data to the NOC central monitoring dashboard. For example

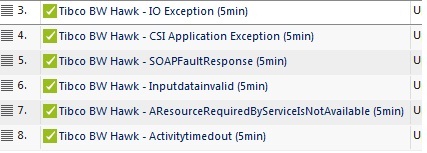


Figure . BW Hawk Alerts

When one of these sensors reaches a predefined threshold, an email alert message is sent to the Asurion Tibco SQA team. Hawk also sends the NOC team notification via their Spectrum Dashboard. The dashboard automatically creates a service desk ticket that NOC may promote to a CIRT. Each Hawk sensor includes historical views and instructions on escalating alerts.

**Also See**

“Monitoring with Hawk” in this document.

#### Administrator Alerts

The Asurion Tibco Administrator team can set alerts, in the Tibco Administrator, to send out notifications them when there are changes to a specified feature in a Subscriber Billing System project area. The team can identify the project (for example, Walmart), alert delivery method such as E-mail, conditions triggering the alert, and the frequency of alert messages (immediately, daily, weekly). Other Asurion teams can view alerts but may not modify them unless a request is approved or issued by the Tibco Administrator group

Asurion users turn to the Tibco Administrator Monitoring Management screen to access alerts



Figure . Administrator - Monitoring Management

When a user selects **All Alerts**, the following screen appears.

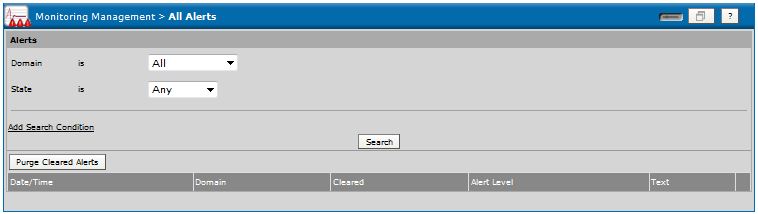


Figure . Monitoring Management Selection

The user select a Domain (for example, “All”), and a State (for example, “Active”), and clicks Search.

For example, the following alert was issued when the service was unable to connect to Global Collect.

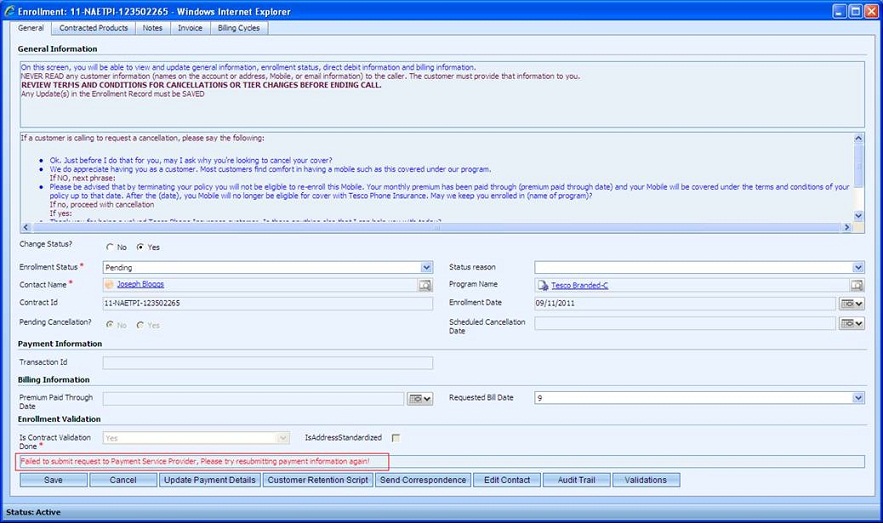


Figure . Sample: Administrator Alert

**Also See**

“Monitoring with PRTG” in this document.

# SQA and Testing

Before the Asurion product is released to production, it is thoroughly tested by our SQA teams. The teams run the Finance Service Gateway through a rigorous suite of testing.

On a system level, SQA testing demonstrates the ability of the Gateway and component gateways and services to properly validate, accept, and process customer data. On a lower level, SQA testing assesses the functions of individual APIs.

At the system level, the Asurion SQA team uses the ARIA Billing Manager to conduct the following tests of the WalMart implementation:

* Smoke test
* Integration test with WalMart and ARIA application
* System test
* Regression test
* End to end test to verify that the system is working properly.

## Billing Manager Tests

The Asurion Software Quality Assurance (SQA) team uses the ARIA Billing Manager application to test the CreateContract Service. In the sample test described below, the Billing Manager is used to conduct System, Smoke, and Regression testing.

An Asurion development team creates a test harness containing sample test data. For example, a test kit could contain customer data (name, address lines, credit card information, and so forth) for a throughput test of the Finance Service Gateway.

In the sample ARIA Billing Manager test shown below, the test subscriber must be successfully enrolled:

* From the Wall-Mart side, the subscriber is successfully enrolled with Asurion via WalMart POS test harness tool.
* At Asurion, the subscriber is successfully enrolled.

### Test: Exercise Subscriber Billing System

The following table lists the actions and expected outcomes for a typical ARIA Billing Manager test of Asurion Subscriber Billing System processes.

**Note:** Typed input is displayed in **bold**. The *filename.csv* is replaced by the actual filename.

|  |  |  |
| --- | --- | --- |
| **Step** | **Description/Input** | **Expected Outcome** |
| 1) Open .CSV file | The .csv file contains xxxxxxx. | The file is opened. |
| 2) Add information to the CSV file. | Enter:   * **MDN number** * **CreateContract details** * **Enrollment date** * **Plan code** | **Note**: See “  ” in this document. |
| 3) Save the file as .CSV |  |  |
| 4) Access the command prompt. | **Run** > **CMD** | The command prompt appears. |
| 5) Enter the command and press Enter. | **walmarttest Test2 *filename.csv***  [ENTER] |  |
| 6) Open the ARIA application. | Start the Aria Billing Manager. | The Aria Billing Manager main page appears. |
| 7) Search for the account by today’s date or the MDN number. | In the Billing Manager:   * Search for the account.by today’s date. * Search for the account again, by the MDN number. | The account appears. |
| 8) Access the Account Overview screen. | Select **Account Overview.** | The Account Overview screen appears. |
| 9) Verify the enrollment details. | Display account details on the Account Overview screen. | Account enrollment details appear on the screen. |
| 10) Access the Client-Defined fields screen. | Select **Client-Defined**. | The Client-Defined fields screen appears. |
| 11) Verify other account details. | Display other account information on the Client-Defined fields screen. |  |
| 12) Access the Master Plan. | Click **Master Plan**. | The Master Plan screen appears. |
| 13) Verify the plan details. | View the plan details. |  |
| 14) Verify the status changes. | Click **Status**. | The status screen appears displaying status the new status. |
|  | Verify the status changes. | Status changes appear on the screen. |
| 15) Verify address details. | Click the **Account Owner**. | The Account Owner screen reappears. |
|  | View the address details. | Address details appear on the screen. |
| 16) Change address details. | On the Account Owner screen, change the address fields. | The changes appear on the screen. |
| 17) Verify the billing dates. | Select **Account Billing Dates**. | The Account Billing Dates screen appears. |
|  | View the billing dates. | Billing dates are displayed. |
| 18) Verify invoices. | Select **Statements & Invoices**. | The Statements & Invoices screen appears. |
|  | View the invoices. | Invoices are displayed. |
| 19) Access the Payments & Credits screen. | Select **Payments & Credits**. | The Payments & Credits screen appears. |
| 20) Collect the payments. | Click **Collect Payments Electronically**. | Account payments are set to be collected. |
| 21) Process the manual refunds. | Click **Refunds**. | Any manual refunds are set to be made. |

#### Sample Test

The following figure shows the command window for a sample test of the Finance Service Gateway.

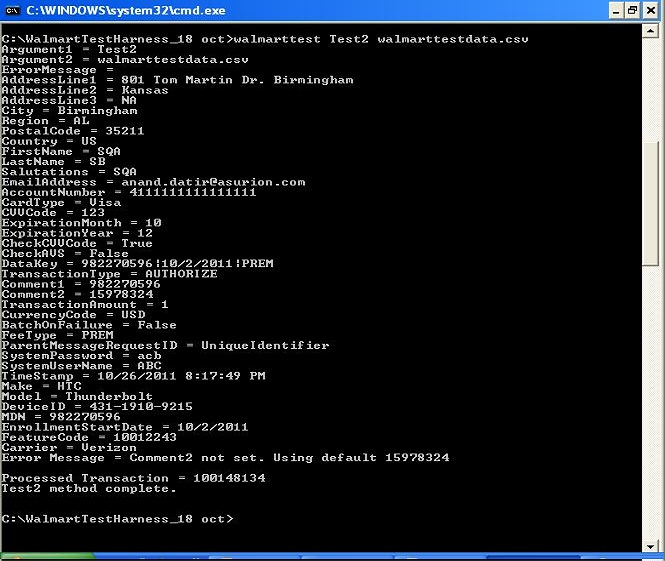


Figure Walmart Test

## Smoke Tests

Smoke tests exercise core operations of the system to verify basic functionality. They are also used to verify the successful implementation of system change requests. Smoke tests determine if further testing is possible and needed. This enables the testing team to avoid wasting time by testing an untestable build.

The result of Smoke tests, successes and failures, are recorded in an SQA database. The following sections describe Asurion Smoke tests.

### Test: Creating an Enrollment

This three-part Smoke testverifies that a WalMart point of sale (POS) user has successfully created an enrollment for handset insurance. A portion of the sample test is outlined in terms of key actions below.

|  |  |  |
| --- | --- | --- |
| **Step** | **Description/Input** | **Expected Outcome** |
| 1) Access the Asurion Billing Manager. | Navigate to the Billing Manager. | The Billing Manager Home page appears. |
| 2) Search for the enrollment. | Using the Billing Manager, search for the enrollment by the subscriber’s name. | The subscriber account is displayed. |
| 3) Verify the enrollment. | View enrollment details. | The enrollment populates all required data :   * First Name * Last Name * Billing Street Address * Billing City * Billing ST * Billing Zip * Mobile Device Number (MDN/Cell Number) * IMEI * Make * Model * Enrolled Date * Credit Card Number * Credit Card Expiration/Credit Card Info * Carrier Name * Program/Service Enrolled or Feature Code |
| 4) Log in to the SQA database. | Access the Hsenroll DB and query for the subscriber enrollment status. View the Active column status. | The enrollment record is displayed and Active = 1. |

**Also See**

“WMT\_SmokeTestscripts\_Sample” at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6DA69BD2-25DE-48A5-A0B8-039239D5C7F8}](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6DA69BD2-25DE-48A5-A0B8-039239D5C7F8%7d).

## Regression Tests

Regression tests are performed to determine the current existing functionality of a system. They are run to verify that new changes deployed to SQA have not broken any system functions. Each time new code is deployed to production, any new functionality is added to the scope of regression tests.

### Test: Failure to Pay Triggers Automatic Cancel Contract Letter

A recent fix of the Subscriber Billing System improved the process for automatically generating subscriber cancel contract letters when a payment is missed. A test script to regression test this feature is summarized below.

|  |  |  |
| --- | --- | --- |
| **Action** | **Description** | **Expected Outcome** |
| Pretest Setup | An enrolled subscriber does not pay by the scheduled date. The Schedule Cancellation date has been reached. | System detects the non-payment and changes the account status in CRM to “Suspended.” |
| 1) Access the system. | User logs onto CRM. | The CRM Home page appears. |
| 2) Search for an enrollment. | Using the Contract Id or MDN, search for the enrollment. | Enrollment records appears. |
| 3) View the enrollment. | Click **View Enrollment**. | Enrollment form shows account information. |
| 4) Verify the enrollment status. | View the enrollment status and reason fields. | Enrollment status field is “Cancel NETF.” The status reason is “Unresolved payment issue.” |
| 5) Go to account billing cycle information. | Click **Billing Cycle** tab. | Billing cycle information appears for the Contract Id with the status “Payment Failure.” |
| 6) Verify the billing cycle status. | View the billing cycle information. | The billing cycle Status is “Exception.” |
| 7) Verify the Note. | View the account information. | The display shows that a Cancellation Note is created indicating the old and new enrollment status and status reasons. |
| 8) Verify that Cancel contract letter processing has started. | View the account information. | The display shows that a Cancel Contract letter has been automatically generated. |
| 9) Access the DB. | Log on to <database>. | User is logged on. |
| 10) Verify that the Cancel Contract letter has been sent to the subscriber. | Query the database:   * Run query using CampaignCorrespondence Select \* from the database object dbo.CorrespondenceQueueCode. * Select \* from the correspondence Queue with ReferenceId = <contractId>. | The records show that the Cancel Contract letter was successfully sent with the name = <contractId>. |

**Also See**

“SubscriberBillingRegressionTestscripts\_Sample” at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6DA69BD2-25DE-48A5-A0B8-039239D5C7F8}](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6DA69BD2-25DE-48A5-A0B8-039239D5C7F8%7d).

## End to End and Integration Tests

Asurion uses similar test scripts for end-to-end and integration testing of Subscriber Billing System projects. For customers in the United States (for example, Walmart, these tests currently exercise two key Finance Services: ProcessNonRefPayments and ProcessRefPayments. For European customers (for example, Bouygues), the entire set of Finance Services are tested end-to-end. The following table summarizes a sample test for the ProcessNonRefPayments finance service in a global context.

### Test: ProcessNonRefPayment Service

The following list of tests together can comprise a total end-to-end service test. Test to:

1. Credit card payment is valid for account “A” with context C1 (specified customer, line-of-business, etc.
2. Credit card payment is valid for account “B” with context C2.
3. Credit card payment is declined for account “A” with context C1.
4. ECheck payment is invalid for account “A” with context C1.
5. Credit card payment is valid for account “A” with context C1.
6. Credit card payment is valid for account “A” with context C3.
7. Credit card payment is declined for account “A” with context C3.
8. ECheck payment is invalid for account “A” with context C3.
9. Credit card payment is valid for account “A” with context C3.
10. Secure3d payment is validly enrolled for account “C” with context C4.
11. Secure3d payment is not validly enrolled for account “C” with context C5.
12. Credit card payment is valid for account “D” with context C5.
13. Credit card is declined for account “D” with context C5.
14. ECheck payment is invalid for account “D” with context C5.
15. BTA payment is valid for account “D” with context C6.
16. ExtendedEcheck payment is valid at the bank for account “F” with context C7.
17. ExtendedEcheck payment is valid at the bank for account “F” with context C7.
18. ExtendedECheck payment has the correct digits for a valid bank for account “F” with context C7.

**Also See**

Sample test code “ProcessNonRefPaymentGlobal\_Sample.cs” at [https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View={6DA69BD2-25DE-48A5-A0B8-039239D5C7F8}](https://oneteam/Teams/OrganizationalTeams/TechnologyProducts/RelationshipProductMgmt/ProductMgmtBackOffice/HRFinanceLegal/SubscriberBilling/Shared%20Documents/Forms/AllItems.aspx?RootFolder=%2FTeams%2FOrganizationalTeams%2FTechnologyProducts%2FRelationshipProductMgmt%2FProductMgmtBackOffice%2FHRFinanceLegal%2FSubscriberBilling%2FShared%20Documents%2FAsurion%20Subscriber%20Billing%20System%20Specs&FolderCTID=0x0120005617996FC3AE2747809B62081EE5F1B4&View=%7b6DA69BD2-25DE-48A5-A0B8-039239D5C7F8%7d).

# Appendix

## A.1 Asurion Subscriber Billing System Error Messages

The .NET and CRM components of the system provide error messages to identify problems arising from functional and user input activities. The key .NET and CRM error messages are described in the following tables.

### A1.1 .NET Error Messages and Meanings

The following errors may occur during .NET service operations.

| **Error Code/Enum Name** | **Error Text** |
| --- | --- |
| INVALID\_FEE\_TYPE = 115 | “Invalid fee type is passed for dopayout.” | |
| INVALID\_PAYMENT\_METHOD = 100 | “Invalid payment method.” | |
| INVALID\_TRANSACTION\_TYPE = 101 | “Invalid transaction type.” | |
| IPP\_SQL\_PROVIDER\_SEARCH\_ERROR = 102 |  | |
| SUBMIT\_PAYMENT\_TO\_PAYPAL = 103 | “Payment submitted to PayPal::<SaleTransaction> Processed.” | |
| RESULT\_CODE\_LOAD\_ERROR = 105 | “Error retrieving 'ResultCodes' node from file’ + resultCodeFileName.” | |
| APPLICATIONMAP\_NOT\_FOUND = 107 | “Application map is not available from AsurionFinance.” | |
| NO\_TRANSACTIONS\_RETURNED = 110 | “No transactions were returned for TransactionId : <transaction\_id>.” | |
| CLIENT\_HASH\_NOT\_VALID = 111 | “ClientHashValue is not valid.” | |
| DATA\_KEY\_NOT\_VALID = 112 | “DataKey is not in correct format.” | |
| INVALID\_FEE\_TYPE = 115 | “Invalid fee type is passed for dopayout.” | |
| OOP\_GET\_INVOICES\_ERROR = 129 | “No premium invoices found for InvoiceId: " + <invoiceId>.” | |
| DUPLICATE\_PROTECTION\_FAILURE = 141 | “The RequestId,’ + <clienthashvalue> + ‘, cannot be processed further due to missing transaction details.” | |
| PAYMENT\_GATEWAY\_MAP\_NOT\_ FOUND = 142 | “Error Loading ‘ + <paymentGatewayMapFileName> + .  File Not Found in ‘ + <sourcefolderpath>.” | |

### A1.2 CRM Error Messages and Meanings

The following errors may occur during CRM service operations.

| **Error Name** | **Error Text** |
| --- | --- |
| ContractProcessingError | “Error occurred in Contract Processing.” |
| TransactionIdNotFound | “Transaction Id not found.” |
| PaymentNotApproved | “Payment Status is NOT APPROVED for Credit Card Premium Account Change.” |
| Error\_Subject | “Error occurred while request processing.” |
| ErrorValidationGates | “Error while updating some of the validation gate.” |
| ErrorWhileCallingCompletePending Contract | “An error occurred while calling Complete Pending Contract.” |
| ErrorWhileCallingCreateContract | “An error occurred while calling Create Contract.” |
| StandardErrorMessage | “A system error has occurred.” |
| SqlErrorMessage | “A SQL error has occurred.” |
| ErrorPopulatingCountriesRegionData | “Error populating countries and region data.” |
| ExceptionInBindingStates | “Exception in binding states.” |
| ExceptionInBindingCountries | “An error occurred in binding countries.” |
| PaymentInformationDeclined | “Payment information was declined.” |
| ErrorInUpdatingPaymentInformation | “Error has occurred while updating payment information. Please try again.” |
| ErrorInUpdatingContact | “Error has occurred while updating contact.” |
| ErrorMessageFromGC | “Failed to submit request to Payment Service Provider, Please try resubmitting payment information again!” |

### CreateContract API Error Messages

The following error messages may appear during CreateContract operations.

| **Error Text** | **Meaning** |
| --- | --- |
| Global Collect Errors | GC errors. |
| ”Card expired.” | Expired card. |
| “Not authorized.” | Card is not authorized for this purchase because of the stated error. |
| * “Do not honor.” | Self-explanatory. |
| * “Invalid card number. | “ |
| * “Low funds.” | “ |
| * “Not permitted to card holder.” | “ |
| * “Others.” | “ |
| * “Pick up card.” | “ |
| * “Suspected fraud.” | “ |
| “Referred.” | Referred. |
| “Unable to authorize.” | Fields syntax error. |

# Index

Alert, 11, 12, 13, 17, 18, 19

ARIA Billing Manager, 19, 20

Aria Billing Manager Tests

System Test, 20

BusinessWorks

Hawk, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15

Hawk Console, 7

CalculateTaxesGateway, 5

CancelContract Letter, 23

CIRT, 3, 4, 13, 15, 16, 18

CreateContract Service, 19

CreateContractGateway, 5, 6

CRM, 5, 25

Domain, 18

Error Messages

CreateContract, 26

CRM, 25

NET, 25

Finance Service Agent, 5

Finance Service Gateway, 3, 4, 5, 6, 19, 20, 21

Global Collect, 26

Hawk, 3, 4, 9, 17, 18

Incident Management (IM), 15

NET, 5, 25

NOC, 4, 6, 13, 15, 16, 17

Paessler Router Traffic Grapher (PRTG), 6

ProcessNonRefPayment Test, 24

ProcessNonRefPaymentGateway, 5, 6

ProcessNonRefPayments, 24

ProcessRefPayments, 24

PRTG, 6, 13, 14, 15, 19

Spectrum Dashboard, 18

SQA, 1, 2, 3, 5, 6, 15, 17, 18, 19, 22, 23

SQA Database, 22

Subscriber Billing System

Finance Services, 5

QA Testing, 5

Test

End to End, 19

End-to-End and Integration, 24

Integration, 19

Regression, 19, 23

Smoke, 19, 22

System, 19

Tibco Administrator, 6, 11, 17, 18

TIBCO BusinessWorks (BW), 6

TIBCO Enterprise Management Service (EMS), 6