Avaya Solution & Interoperability Test Lab

Application Notes for Austin Logistics OnQ with Avaya Proactive Contact - Issue 1.0

Abstract

These Application Notes describe the configuration steps required for Austin Logistics OnQ to successfully interoperate with Avaya Proactive Contact.

OnQ 2.0 is a software solution that automates and centralizes campaign as well as list management. OnQ 2.0 uses the Event Service of Avaya Proactive Contact 4.0 to extract job statistic event information. OnQ 2.0 can retrieve the call results data for a job either in a batch mode or using event services. The test configuration consisted of Avaya Communication Manager 5.0 (running on Avaya S8710 Servers with a MCC1 Media Gateway) and Avaya Proactive Contact 4.0.

Information in these Application Notes has been obtained through DevConnect compliance testing and additional technical discussions. Testing was conducted via the DevConnect Program at the Avaya Solution and Interoperability Test Lab.

1. Introduction

These Application Notes describe the compliance-tested configuration utilizing Avaya Proactive Contact 4.0 (Avaya PC4) and Austin Logistics OnQ 2.0 (OnQ). OnQ is a software solution that automates and centralizes campaign as well as list management. OnQ utilizes customer data such as number and type of enterprise relationships to determine calling schedules. Unlike traditional call strategies that are limited to customer phone number and call history, these strategies use a new level of intelligence to regulate recall of high valued customers.

1.1. Interoperability Compliance Testing

This interoperability compliance testing covers feature functionality and serviceability. Feature functionality focuses on verifying that Austin Logistics OnQ can successfully send call records to Avaya PC4 based on the workload and on-going call results. Serviceability testing verified that OnQ recovers from cable disconnects/re-connects and stopping and starting of Avaya PC4 Event Service. OnQ service and server reboots were not performed as OnQ service needs to be started manually.

At the start of a day, OnQ receives a file of accounts to be called from a host system. This list is loaded into OnQ database and accounts are sent to Avaya PC4 throughout the day using Secure FTP (SFTP) on an as-needed basis. After each send, OnQ uses Avaya PC4 Event Service to poll for status to determine the number of records to be sent in the next cycle.

The OnQ integration with Avaya PC4 requires custom development scripts on Avaya PC4 from Avaya Professional Services. The following steps are performed for this integration to work:

- OnQ sends the call records file to Avaya PC4 using SFTP.
- Avaya PC4 runs scripts to pick up the call record file and appends to the OnQ infinite calling lists.
- OnQ uses Avaya PC4 job statistics event to determine when to send the next batch of call records.
- Avaya PC4 infinite job processes the records and generates a result file at a specified interval.
- OnQ picks up the result file to reprioritize and rework the call records and distribution based on the workload and call results.

1.2. Support

If technical support is required for the Austin Logistics OnQ solution, then contact Austin Logistics Technical Support. Full details are available at https://www.AustinLogistics.com.

2. Reference Configuration

Figure 1 depicts an overview of the Austin Logistics OnQ integration to Avaya Proactive Contact. The configuration consists of a pair of redundant Avaya S8710 servers, an Avaya MCC1 Media Gateway, Avaya IP Telephones, an Avaya Proactive Contact System Cabinet, agent workstations, and the OnQ server.

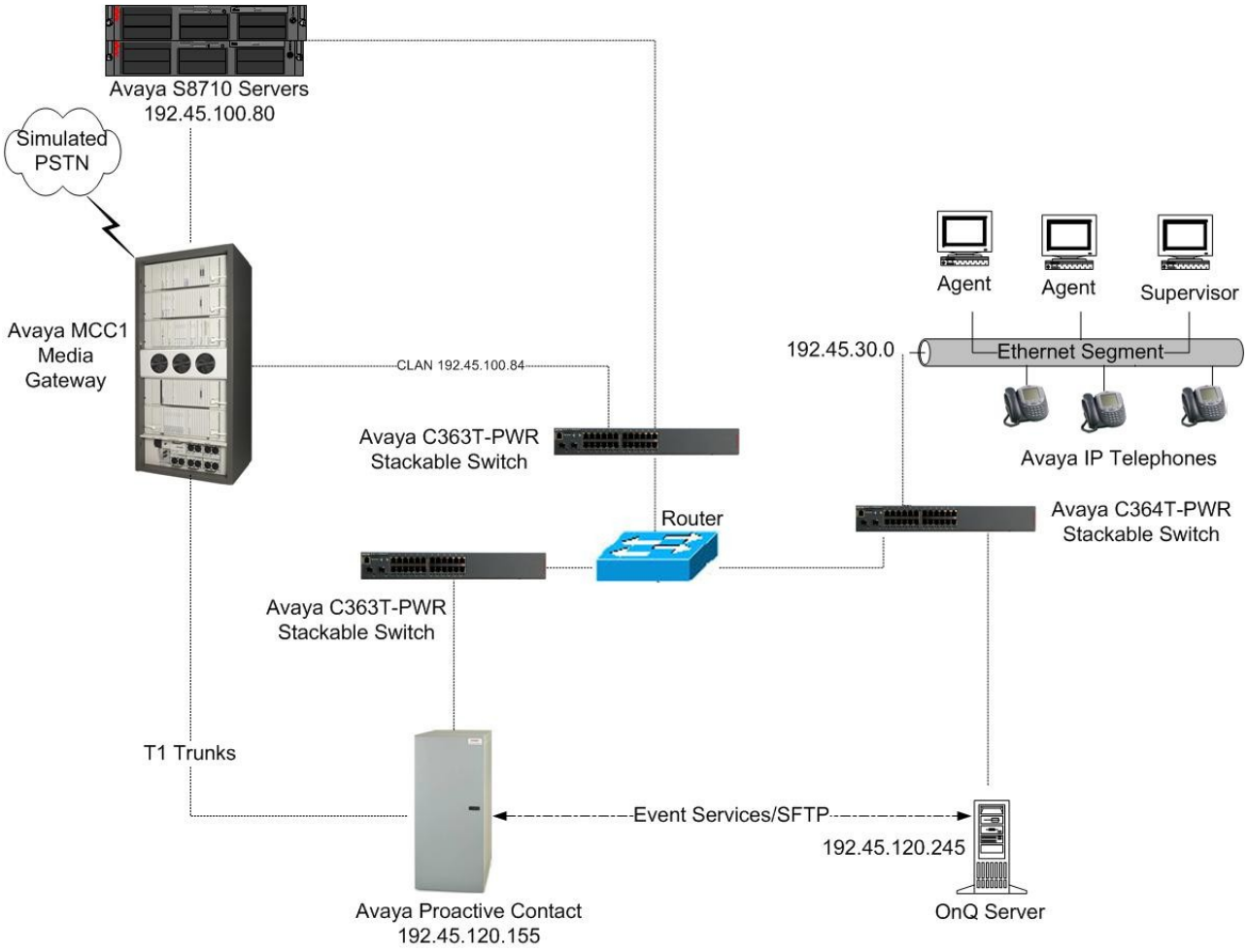


Figure 1: Avaya Proactive Contact and Austin Logistics OnQ Integration

3. Equipment and Software Validated

The following equipment and software were used for the tested configuration:

Equipment	Software
Avaya Proactive Contact installed on HP Proliant DL385G2 Server using Digital PG230 Switch	Avaya Proactive Contact 4.0.1 SP 1, Build 105
Avaya MCC1 Media Gateway with Avaya S8710 Servers	Avaya Communication Manager 5.1.1 (R015x.01.1.415.1)
TN464F DS1	Version 16
Avaya 4610SW IP Telephones (H.323)	2.8
Avaya C364T-PWR Converged Stackable Switch	4.5.14
Austin Logistics Linux Server	OnQ 2.0

4. Configure Avaya Communication Manager

The Avaya Communication Manager to Avaya Proactive Contact configuration is outside the scope of these Application Notes and should already be operating properly [2].

5. Configure Avaya Proactive Contact 4.0

These Application Notes assume that the interface with Avaya PC4, Avaya S8710 Servers and Avaya Communication Manager has been configured and is operational. The following features are already configured on Avaya PC4.

- Outbound Calling
- Infinite Job Feature

Avaya Professional Services will install custom scripts and modify configuration files on Avaya PC4 for the OnQ integration. Austin Logistics will create a call records raw file which will be transferred to Avaya PC4 using SFTP from time to time. The transfer frequency depends on the number of call records remaining in the queue to be dialed. OnQ receives this information by registering with Avaya PC4 Event Service.

5.1. Avaya Professional Services Custom Scripts

Avaya Professional Services custom development on Avaya PC4 is required for this integration. The custom development includes the creation of new scripts and modification of some existing files on Avaya PC4. The following scripts were created:

- **onq_readtape -** Resets the OnQ infinite calling lists each evening.
- **onq_checkfile –** Looks for call records from the OnQ server to append to the OnQ infinite lists.
- **onq_append –** Appends the new call records to the call list.
- **onq_list#.job** – Runs the infinite job call selection process. For this testing **onq_list1** and **onq_list2** jobs were created to run against **list1** and **list2**.

- **onq_extract –** Creates the **results.dat** file every 10 minutes by running a PC Analysis extract on each OnQ calling list.

Note: Avaya PC4 requires at least one record in the call list to start an infinite job.

5.2. OnQ Call Records Raw File

OnQ needs to create call records raw files to send to the public/onq folder on Avaya PC4. The following files from Avaya PC4 are needed to help Austin Logistics create the call records raw file.

- /opt/avaya/pds/lists/list1.app/list1.dnld – This file contains the raw file configuration section. This configuration section indicates where the raw file from OnQ is expected to be located on Avaya PC4 and what list name will be generated.

```
#------------------------CONFIGURATION (formerly .conf)-------------------:
CONF_START:
RECSIZE:160
BLKSIZE:1600
BLKSREAD:10
TERMINATOR:
CHARSET:ASCII
CASE:UPPER
#OnQ is expected to place file in the directory listed below (rcvfile1.raw was used for this compliance testing)
TAPEDEV:$VOICEDIR/xfer/public/onq/rcvfile1.raw
TAPENAME:list1 (Call list record file generated from the raw file)
FILEFORMAT:FIXED
RECORDDELIMITER:
FIELDDELIMITER:
QUALIFIER:"
CONF_END:
```

Figure 2: Configuration File for Raw file

- /opt/avaya/pds/lists/list1.app/list1_dn.dict - This is the dictionary file which includes the fields and their position in the raw file generated by OnQ for processing. **Figure 3** shows the sample dictionary file for the raw file.

```
DEFL:SYSNUM:4:C:1:SYSNUM::
DEFL:PRIN:4:C:5:PRIN::
DEFL:CCODE:3:C:9:CCODE::
DEFL:ACCTNUM:16:C:12:ACCTNUM::
DEFL:NAME1:26:C:28:NAME1::
DEFL:NAME2:26:C:54:NAME2::
DEFL:CBFLAG:1:C:80:CBFLAG::
DEFL:PHONE2:10:C:81:PHONE2::
DEFL:AREA2:3:C:81:AREA2::
DEFL:PHONE1:10:C:91:PHONE1::
DEFL:AREA:3:C:91:AREA::
```

```
DEFL:EXTERNAL:1:C:101:EXTERNAL::
DEFL:INTERNAL:1:C:102:INTERNAL::
DEFL:BALANCE:9:$:103:BALANCE::
TFORM:####### ##
DFORM:$$$$$$#.##
DEFL:CREDLINE:7:$:112:CREDLINE::
TFORM:#######
DFORM:$$$$$##
DEFL:DELQUENT:9:$:119:DELQUENT::
TFORM:####### ##
DFORM:$$$$$$#.##
DEFL:DAYS:3:N:128:DAYS::
DEFL:PAYDAY:10:D:131:PAYDAY::
TFORM:CCYY MM DD
DFORM:CCYY/MM/DD
DEFL:PAYAMT:7:$:141:PAYAMT::
TFORM:##### ##
DFORM:$$$$#.##
DEFL:ZIPCODE:5:C:148:ZIPCODE::
DEFL:BEHSCORE:3:C:153:BEHSCORE::
```

Figure 3: Raw File Layout

An example of the call records raw files is shown in **Figure 4**. It contains three customer records that can be appended to an existing calling list.

```
1072920040143022098601015 46JOHN DOEJOHN DOE000000000002033234562FD00000000000000
0000000000008790010800000000690286011107292004014302209860095714JOHN DOEJOHN DOE
020335984772033483589FD000000000000000000000000000578903100000000069028601110729 2
00401430220986009377 6JOHN DOEJOHN DOE100000000002032459424FD0000000000000000000 0
0000004889120800000000644386011
```

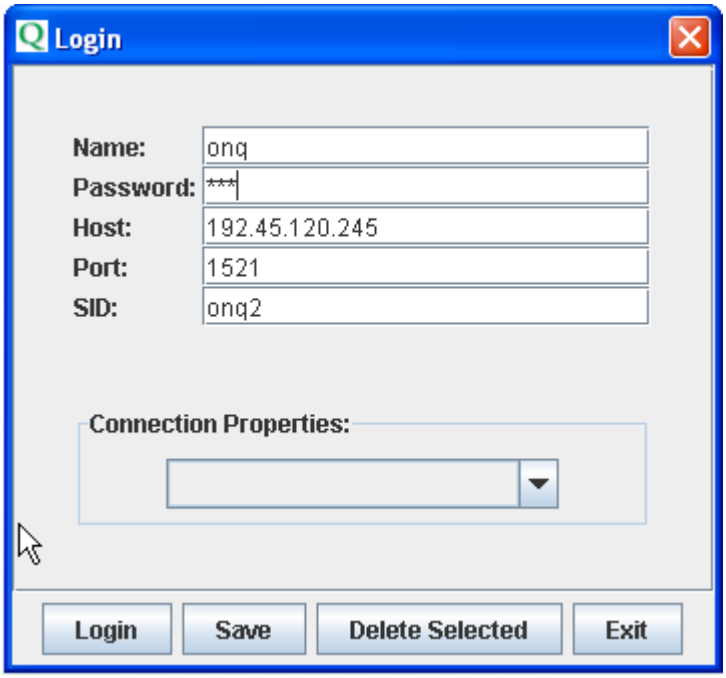
Figure 4: Sample Raw File

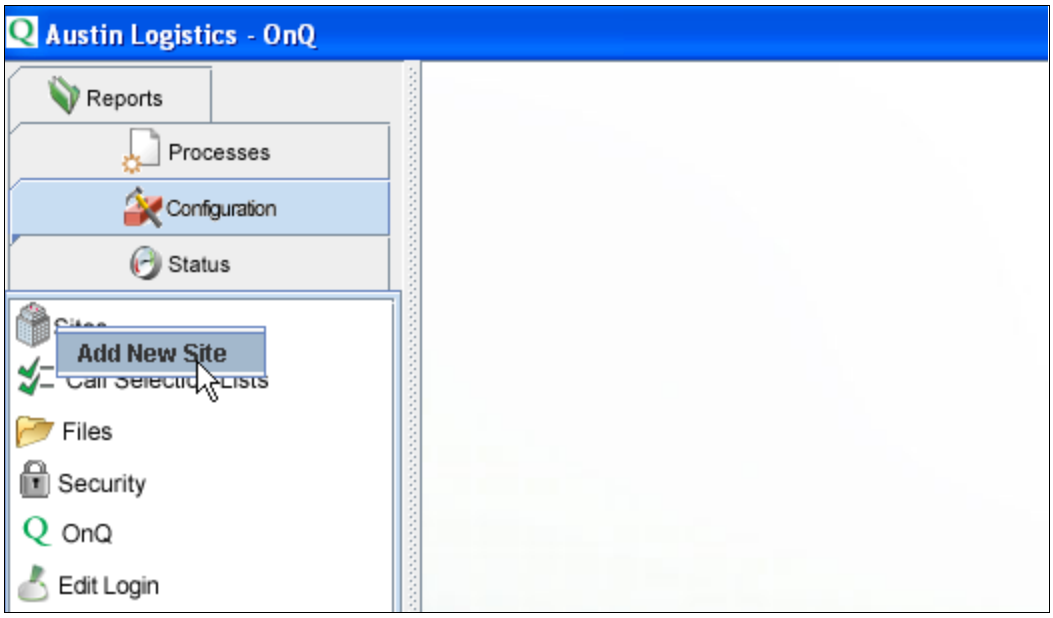
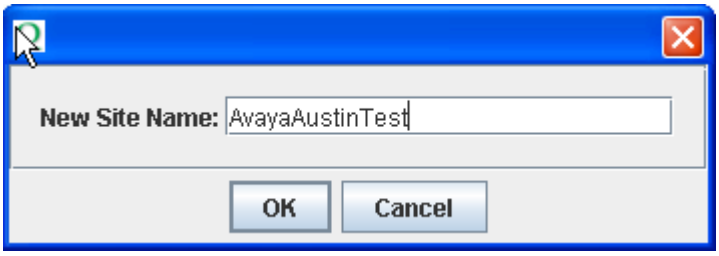
5.3. Avaya PC4 SFTP

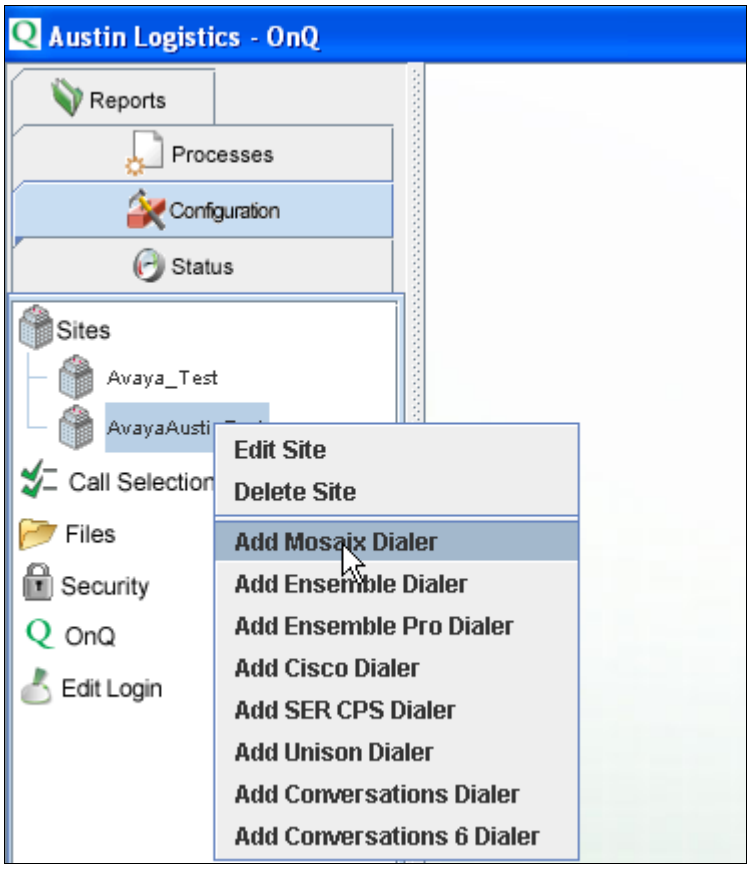
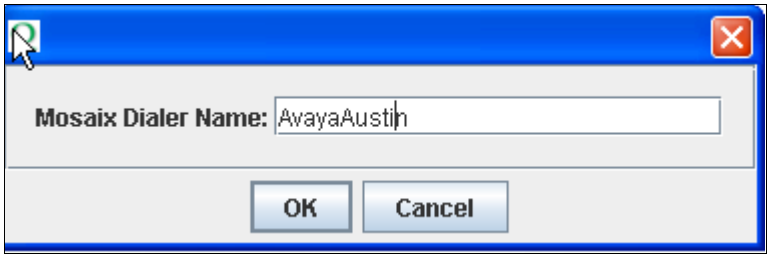
SFTP is used by OnQ to send the call record raw files and retrieve the results file.

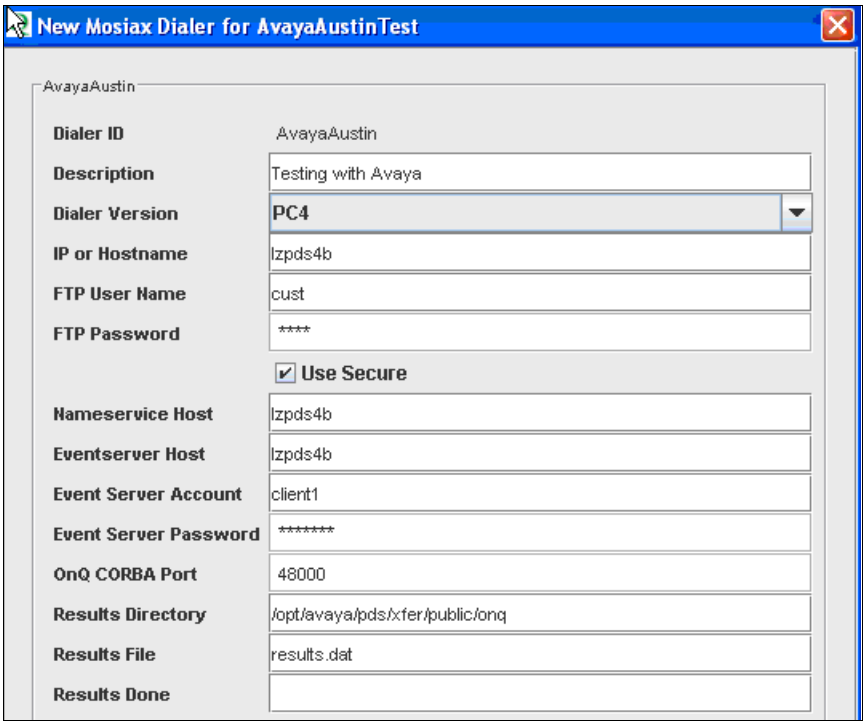
6. Configure Austin Logistics OnQ

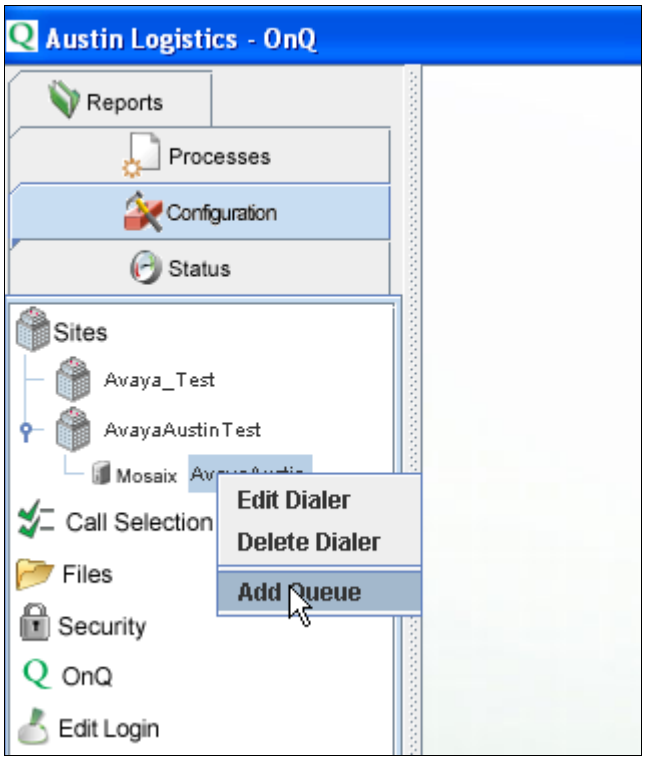
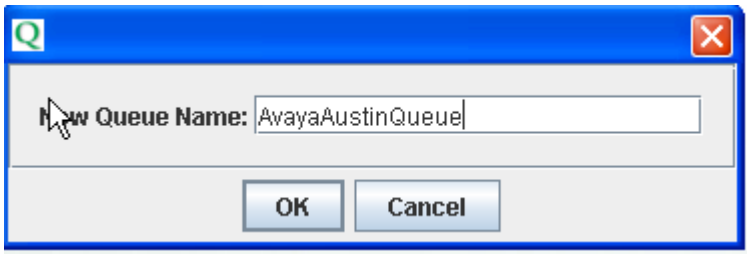
These Application Notes assume the Austin Logistics OnQ software has been installed successfully. Values are entered only in fields which need to be configured. For all other fields, default values are used.

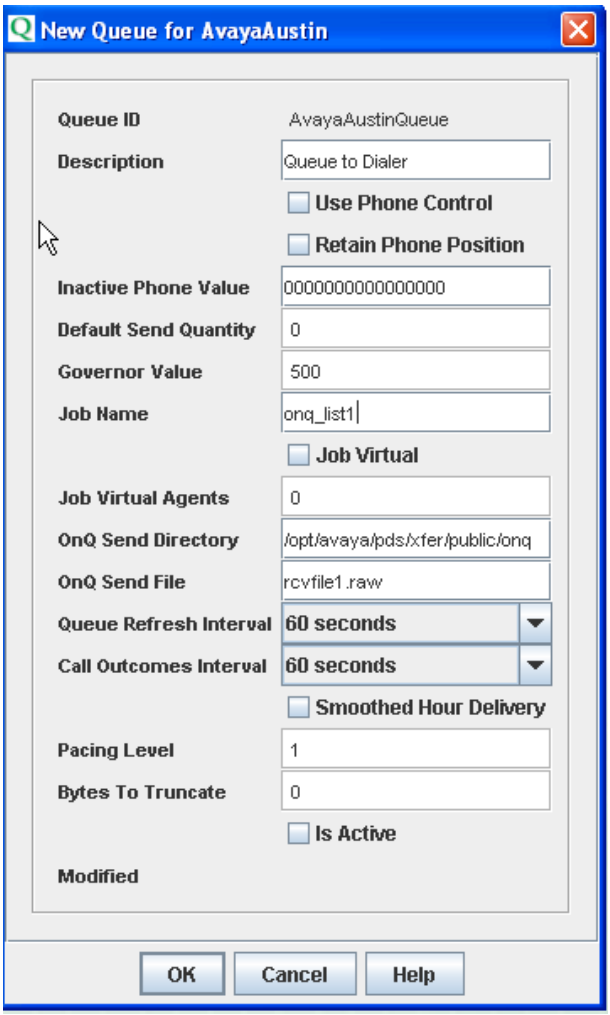
Step	Description
1.	Access the OnQ administration interface by clicking the **OnQ** icon created on the server during the installation procedure and configure as follows: • **Name** – Name of the admin user configured on OnQ server. • **Password** – Password for the user configured on OnQ server. • **Host** – IP address of the OnQ server. • Click **OK** and close the next popup window [not shown]. 

Step	Description
2.	At the **Austin Logistics – OnQ** window, select the **Configuration** tab and right click on **Sites.** Select **Add New Site** from the drop down list and enter any descriptive string in the **New Site Name** field at the next popup window and click **OK**.  

Step	Description
3.	Expand the **Sites** tab and right click on the site added in previous step. Select **Add Mosaix Dialer** from the drop down list and enter any descriptive string in the **Mosaix Dialer Name** field at the next popup window and click **OK**.  

Step	Description
4.	At the **New Mosiax Dialer** window, configure as follows: • **Description** – Enter any descriptive string. • **Dialer Version** – Set to **PC4** for testing with Avaya Proactive Dialer 4.0 from the drop down list. • **IP or Hostname** – Enter Avaya PC4 host name. Do not use IP address. • **FTP User Name** – Set to user name created for SFTP. • **FTP Password** – Enter a valid password. • **Use Secure** – Check this field to interface with secure IIOP on Avaya PC4. • **Nameservice Host** – Enter Avaya PC4 host name. Do not use IP Address. • **Eventserver Host** – Enter Avaya PC4 host name. Do not use IP Address. • **Event Server Account** – Enter the user name created on the Avaya PC4 for getting the corba events. • **Event Server Password** – Enter a valid password • **Results Directory** – This is where the **results.dat** file is picked from after **onq_extract** script retrieves the records from the call record list. Set to **/opt/Avaya/pds/xfer/public/onq** for this compliance test. • **Results File** – Set to **results.dat** generated by **onq_extract** script. • Click **OK** [not visible]. 

Step	Description
5.	Right click on the Mosaix Dialer added in **Steps 3-4** and select **Add Queue** from the drop down list. Enter any descriptive in the **New Queue Name** field at the next popup window and click **OK**.  

Step	Description
6.	At the **New Queue** window, configure as follows. • **Description** – Enter any descriptive string. • **Job Name** – Enter the jobname configured on Avaya PC4 using **onq_list**<*calling list #*> format. Set to **onq_list1** for this compliance test. • **OnQ Send Directory** – This is where the call records file is picked from after OnQ transfers the file to the dialer. **onq_checkfile** script picks up this file and appends the call records to the list. Set to **/opt/Avaya/pds/xfer/public/onq** for this compliance test to match the directory name in **Section 5.2**, **Figure 2**. • **OnQ Send File** – Enter the name of the raw file containing the call records using the **rcvfile**<*calling list #*>**.raw** format. Set to **rcvfile1.raw** for this compliance test to match the filename in **Section 5.2**, **Figure 2**. • Click **OK**. 
7.	From the Linux server start the OnQ by executing **onqctl –start onq –debug 2** in **$ HOME/bin** directory.

7. General Test Approach and Test Results

All feature functionality test cases were performed manually to verify proper operation. The general test approach entailed:

- Establishing connectivity between Austin Logistics OnQ and Avaya PC4.
- Verifying job statistics events are received by OnQ from the Event Service on Avaya PC4.
- Verifying files can be sent and retrieved using SFTP on Avaya PC4.
- Verifying call records can be added to the infinite job's calling list running on Avaya PC4 using the list management feature of OnQ.
- Verifying Call Data and Agent Data events are received by OnQ and OnQ is able to report the Completion codes accurately.

All feature and serviceability tests passed. OnQ successfully sends the call records to Avaya PC4 throughout the day while the infinite job is running. OnQ polled Avaya PC4 for the results.dat file and the job statistics in order to determine how many records to send next and when to send them.

For serviceability testing, OnQ was able to resume sending the call records after restoration of connectivity to the Avaya PC4 server from network disconnect/re-connect and Avaya PC4 stop and restart.

The following observations were obtained from testing:

- The OnQ Queue Status screen does not show the updated status of the Event Service connection to Avaya PC4 when the connection is down. Currently the administrator will see a static screen with no indication that the connection to Avaya PC4 is down on the Queue Status screen. The job state field will continue to display **active**. The connection failure is only recorded to the log files.

8. Verification Steps

8.1. Avaya Verification

Step	Description
1.	Execute the "**netstat –a"** from the command prompt on the Avaya PC4 server to verify communication between Avaya PC4 and the OnQ. The result below shows that OnQ at 192.45.30.233 is connected to the Avaya PC4 (lzpdsb).

```
$ netstat -a
Active Internet connections (including servers)
Proto  Recv-Q Send-Q  Local Address        Foreign Address        (state)
tcp    0      0       lzpdsb.57209         192.45.30.233.49370    ESTABLISHED
tcp    0      65      lzpdsb.agent         192.45.30.242.1309     ESTABLISHED
```

Step	Description
2.	Execute the "**enclient $NS –J"** command from Avaya PC4. The job statistics results will be shown. The highlighted fields are the statistics used by Austin Logistics OnQ. Verify the data in these fields to match the data in **Section 7.2**.
	Static Job Data: **jobName ="onq_list1"** callingList ="lzpds-list1" recordSelectionFil="onq_list1" phoneStrategyFile ="infinite_strategy" jobStartTimeStamp =2009/1/20-10:48:16 jobEndTimeStamp=Null **jobNumber = 256** jobSlot = 1 jobType ='O' linesAssigned = 4 totRecsToCall = 20 Dynamic Job Data: cruiseControl = 0 desiredServiceLevel = 0.990000, connectTolerance = 1 servicedCalls = 18, offeredCalls = 19 **runningHitRate = 84 currentHitRate = 72** outbTotalQueCalls = 1 outbOutQueCalls = 0 outbAverageQueTime= 16 outbTotalQueTime= 16 recordsCalled = 8 **recordsAvailable= 2** recordsRecalled= 1 activeStatus=1 setupFinished=1 inShutdown=0 noMoreCalls=0 Outb Stats: outCallsPlaced = 22 outRecallsPlaced= 1 **outCallsAnswered = 30** outCallsInWait = 1 outCallsWorked = 22 outIdleCount = 22 outWaitQueueTime= 16 outWorkTime = 382 outIdleTime = 401 outTalkTime = 287 outUpdateTime = 95 Agent Counts: I=0 **O=2** B=0 M=0 P=0 A=0 Comp Codes: code= 11 callType=O racCode=0x2 count=11 code= 15 callType=O racCode=0x2 count=9 code= 17 callType=O racCode=0x2 count=5 code= 49 callType=O racCode=0x2 count=1 code= 89 callType=O racCode=0x0 count=24
3.	Verify that the raw file to be processed by Avaya PC4 is being put in the directory configured in **Section 6**, **Step 6** and the results file is being put in the directory configured in **Section 6**, **Step 4** to be picked up by OnQ.

8.2. OnQ Verification

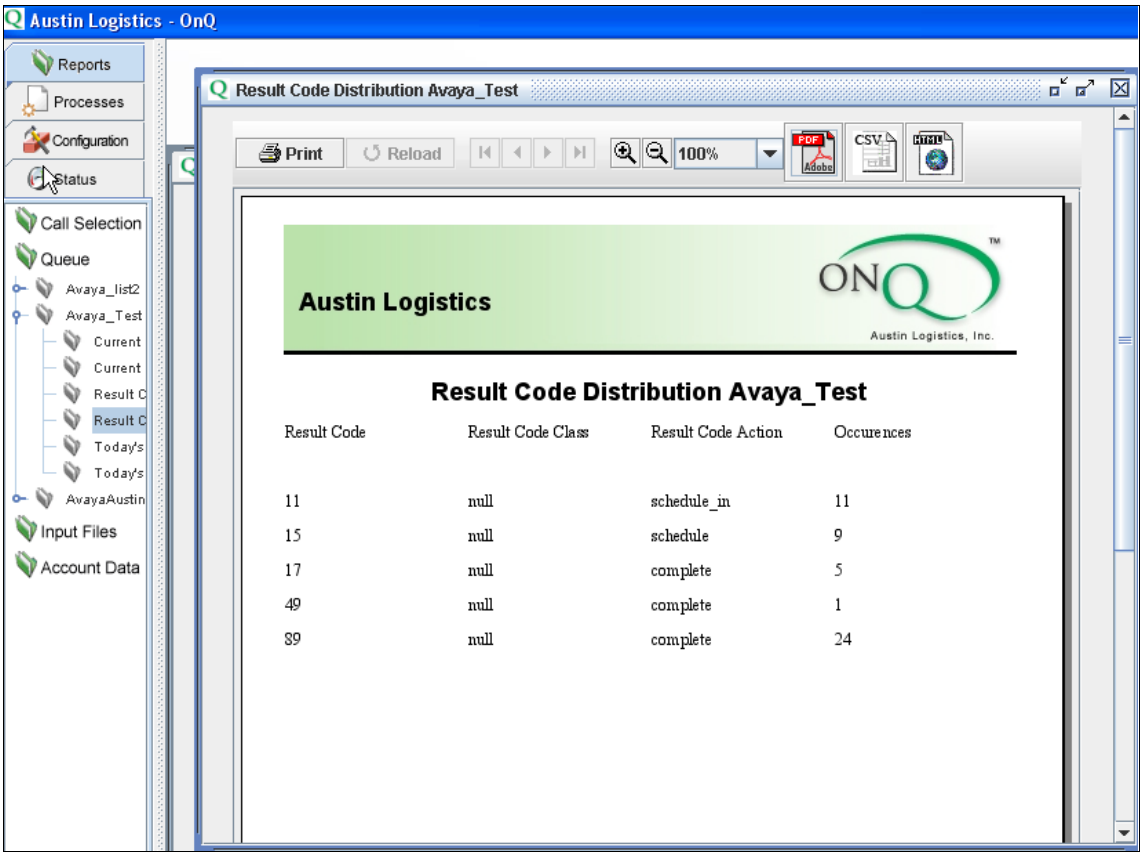
The following steps can ensure that the communication between OnQ and Avaya PC4 are working.

<table>
<tr><th>Step</th><th>Description</th></tr>
<tr><td>1.</td><td>To verify that OnQ server is running and connected to Avaya PC4, Open the log AvayaAustin_es_client.log in $HOME/install/local/logs directory and verify that Avaya PDS Corba ready and waiting and Connection appears to be active are displayed in this file. The file name prefix is AvayaAustin, the dialer name configured in Section 6, Step 3.

Jan 22, 2009 2:49:19 PM com.ali.onq.avayapdscorba.AvayaPDSCorba getDialerEventServer
INFO: got DialerEventService interface
Jan 22, 2009 2:49:19 PM com.ali.onq.avayapdscorba.AvayaPDSCorba runPDSClient
INFO: start(): logging into dialer
Jan 22, 2009 2:49:19 PM com.ali.onq.avayapdscorba.AvayaPDSCorba runPDSClient
INFO: Avaya PDS Corba ready and waiting.
Jan 22, 2009 2:49:23 PM com.ali.onq.avayapdscorba.EventClientImpl jobStatNotify
FINE: Job Stats notification
INFO: Connection appears to be active.
Jan 22, 2009 8:26:37 AM com.ali.onq.avayapdscorba.EventClientImpl systemStatNotify
FINE: System Stats notification
Jan 22, 2009 8:26:37 AM com.ali.onq.avayapdscorba.EventClientImpl jobStatNotify</td></tr>
<tr><td>2.</td><td>To verify job statistics on OnQ server, open and view the AvayaAustin_es_client.out.jsd in $HOME/install/data directory. The file name prefix is AvayaAustin, the dialer name configured in Section 6, Step 3.

The following fields from the Avaya PC4 Event Service correspond to the file on OnQ server and were verified both on Avaya Dialer and OnQ side:

<table>
<tr><th>Avaya PC4 Event Service</th><th>OnQ Status Queue</th></tr>
<tr><td>jobName</td><td>name</td></tr>
<tr><td>jobNumber</td><td>number</td></tr>
<tr><td>runningHitRate</td><td>runrate</td></tr>
<tr><td>currentHitRate</td><td>currate</td></tr>
<tr><td>recordsAvailable</td><td>avail</td></tr>
<tr><td>outCallsAnswered</td><td>outcon</td></tr>
<tr><td>Agents Counts</td><td>agentcount</td></tr>
</table>
</td></tr>
</table>

Step	Description
3.	Open OnQ client and navigate to **Report->Result Code Distribution** to verify that the completion codes match the completion codes on Avaya PC4 job statistics event. 
4.	Verify that the result file is picked up from Avaya PC4 and put on OnQ server in $HOME/install/data directory. Open **AvayaAustingQueue_results.out** to verify the contents of the file match the results file on Avaya PC4. The file name prefix is **AvayaAustinQueue**, the dialer name queue configured in **Section 6**, **Step 5**.

9. Conclusion

These Application Notes describe the required configuration steps for Austin Logistics OnQ 2.0 to successfully interoperate with the Event Service and SFTP of Avaya Proactive Contact 4.0 list management. Custom development work was needed on Avaya PC4 by Avaya Professional Services to integrate this solution. Functionality and serviceability were successfully validated. The configuration described in these Application Notes has been successfully compliance tested.

10. Additional References

The following documents may be found at http://support.avaya.com:

[1] *Administrator Guide for Avaya Communication Manager*, Document ID 03-300509, Issue 4, January 2008

[2] *Implementing Proactive Contact 4.0*, May, 2008

[3] *Administering Avaya Proactive Contact 4.0 (Linux-based Interface)*, January 2008

Austin Logistics product documentation is available on request from https://www.AustinLogistics.com.

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya DevConnect Program at devconnect@avaya.com.