

TECHNICAL WHITE PAPER

CM Pump-Up and Data Flow in Avaya IQ

Date: Apr 5, 2015

Author: Vinit Dave and Tosif Shaikh

Abstract

When the CM link to IQ is established, one of the primary activities is to transfer data from CM to IQ, this is called the Data Pump-up process. The data which is transferred includes administrative, operational and names data. This white paper provides an overview of the data transfer activities which occur during the Data pump-up process. It also provides information on troubleshooting cases where data transfer did not work.

Terminology

ssh	Secure Shell. Used for Secure Remote Access Connection.
sftp	Secure File Transfer Protocol
ftp	File Transfer Protocol
TCP	Transmission Control Protocol
JMS	Java Messaging Services
HDR	Historical Dimensional Recorder - Processes administrative data and names collected from one or more external data sources and records it in the Historical Data Store.
RDR	Real-time Dimensional Recorder - Processes administrative data and names collected from one or more external data sources and records it in the real time database.
CMIT	Communication Manager Input Translator
Admin Recorder	Receives admin message from CMIT and forwards it to DMM
DMM	Device Management Model
HOT	Historical Output Translator
HEP	Historical Event Processor
ROT	Real-time Output Translator
REP	Real-time Event Processor
Administrative data	Data about how routing points, agent IDs, splits, trunks, skills, and so on are administered. Also called “RFTB” (Request Full Translation Block).
Operational state data	Data that reflects the current state of all active agents. Also called “RLTB” (Request Logon/Status Translations).
Contact data	Data that records call center contact information. For example, the telephone number of a caller.
Names data	Contains names of CM entities, such as agents, routing points, queues, and trunk groups, and vectors.
PE	Processing Element
CM Message Parser	The CM Message Parser decodes Communication Manager events from their original binary hexadecimal format into a format suitable for the Communication Manager Input Translator (CMIT).
DSS	Distributed System Services - core services library that provides internal inter-process communications.

Synchronization:

Avaya IQ synchronizes translation data in the Avaya IQ databases with the translation data in the data source databases to keep Avaya IQ reports current.

Synchronization Types:

Avaya IQ has the following types of synchronizations: Initial, Subsequent and Names. The following sections provide details on each type of synchronizations.

1. Initial

Initial synchronization occurs during implementation when the Avaya IQ databases are getting built. During initial synchronization, report users cannot access the user interface or generate reports for up to 20 minutes. This type of synchronization is commonly referred to as “pump-up”. Initial synchronizations involve more activity than subsequent synchronizations. All administration data from the data sources are synchronized. In addition, all Role Based Access Control (RBAC) permissions are synchronized, which includes queues, agents, devices, processes and routing points.

2. Subsequent

This type of synchronization occurs when the existing databases need to be updated because a change occurred in Avaya IQ or in the data source. Avaya IQ monitors translation data and initiates synchronizations when it discovers changes in Avaya IQ or in the data sources. On large data source systems, synchronization takes about 5 minutes. Synchronization times are significantly shorter on smaller systems. Even though report users still have access to Avaya IQ reports during synchronization, reports do not contain the updated data until synchronization is complete. Report users also cannot run reports on new Communication Manager entities for up to 30 minutes. In subsequent synchronizations, all administration data and operational state data are synchronized. Only changes made to RBAC permissions are synchronized. RBAC permissions are processed out of a queue that is separate from the data flowing from the data sources. Avaya IQ also initiates the synchronization after network interruptions. After the network is restored, if Avaya IQ detects any changes in the data source, administration and operational state data are synchronized. If Avaya IQ does not detect any changes, only operational state data is synchronized.

Note:

Subsequent synchronizations between CMS and Communication Manager work the same as subsequent synchronizations between Avaya IQ and Communication Manager.

3. Names

Avaya IQ performs the names data synchronization with Communication Manager after the initial or subsequent synchronization is complete. The names synchronization can take some time to complete. Currently, the names in Communication Manager translation data can synchronize with Avaya IQ but not vice versa.

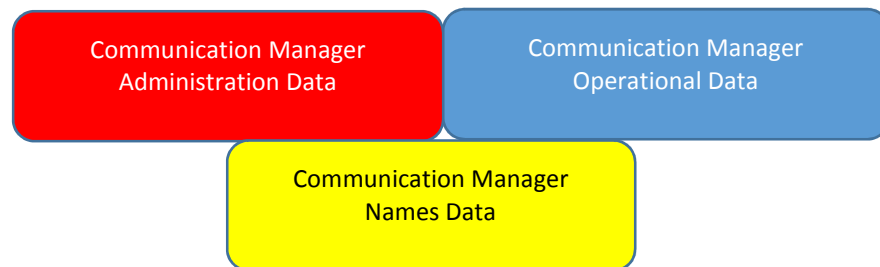
What data gets synchronized?

From Communication Manager

The following data gets synchronized:

- **Request Full Translation Block (RFTB)** message is sent by IQ to request a 'full translation pump up, also referred' to as administrative data or Admin pump up.
- **Request Logon/Status Translations (RLTB)** message is sent by IQ to request updates of agents that are currently logged in, and their current state, also referred to as operational state data or Operational (OPER) pump up.
- **Request names (REQNAM20)** message is sent by IQ to request names of CM entities, also referred to as names data or Names pump up.

The following diagram describes how synchronization between Avaya IQ and Communication Manager is processed.



1. Communication Manager Administration data synchronization starts.
2. Once all Communication Manager Administration messages are stored in the queue, Communication Manager names data synchronization starts and Communication Manager operational data synchronization is requested. Names data synchronization is non-blocking and is interrupted until operational synchronization completes.
3. When Communication Manager Operational data synchronization completes, names data synchronization resumes. During the continuing names data synchronization, contact data (standard CM traffic data) begins flowing from the CM to IQ and is persisted into the database.

Definitions:

PE Event Processor

Manages event processing and processes normalized events from the PE CM Adapter and PE PD Adapter. This Records real-time data to the Real Time Data Store.

Container: Data Processing JBoss.

If this PE fails: Data is lost. If this PE shuts down, the PE CM Adapter and the PE PDS Adapter also shut down. If a remote PE Event Processor stops running, the PE CM Adapter and the PE PDS Adapter buffer some data, but eventually drop the overflow data.

PE Recorder

This processing element records fact data to the Historical Data Store.

Container: Data Processing JBoss.

If this PE fails: Historical data does not flow into the Historical Data Store, thereby affecting the data in the historical reports. Avaya IQ buffers data before it gets recorded to the database host if the PE Recorder has stopped. Data buffering saves the events being sent by a data source to a buffer on a disk or in memory before the system sends the events over the network. If the network fails or if there is a failure on the database host, the events continue to accumulate in the buffer until the failure is corrected or until the disk buffer becomes full. When the network is restored, the system sends the events in the buffer before the new events that come from the source. Every remote site must use data buffering.

PE CM Adapter

This object monitors the CMAadapter. The CMAadapter implements the protocol that manages the communications link between Communication Manager and Avaya IQ. More specifically, the CMAadapter prepares the data stream for event processing. One CMAadapter monitors one Communications Manager. Your Avaya IQ system must have one or more CMAadapters.

Container: Data Processing JBoss and Data Collection JBoss.

If this PE fails: Call event data between Communication Manager and Avaya IQ is lost.

PE HDR Entity Monitor

This element monitors the HDR. The Historical Dimensional Recorder (HDR) processes administrative data and names collected from one or more external data sources and record it in the Historical Data Store.

Container: Admin JBoss.

If this PE fails: Historical administration data is not recorded. Historical administration data includes initial synchronization information and updates to names of agents, queues, routing points, and so on.

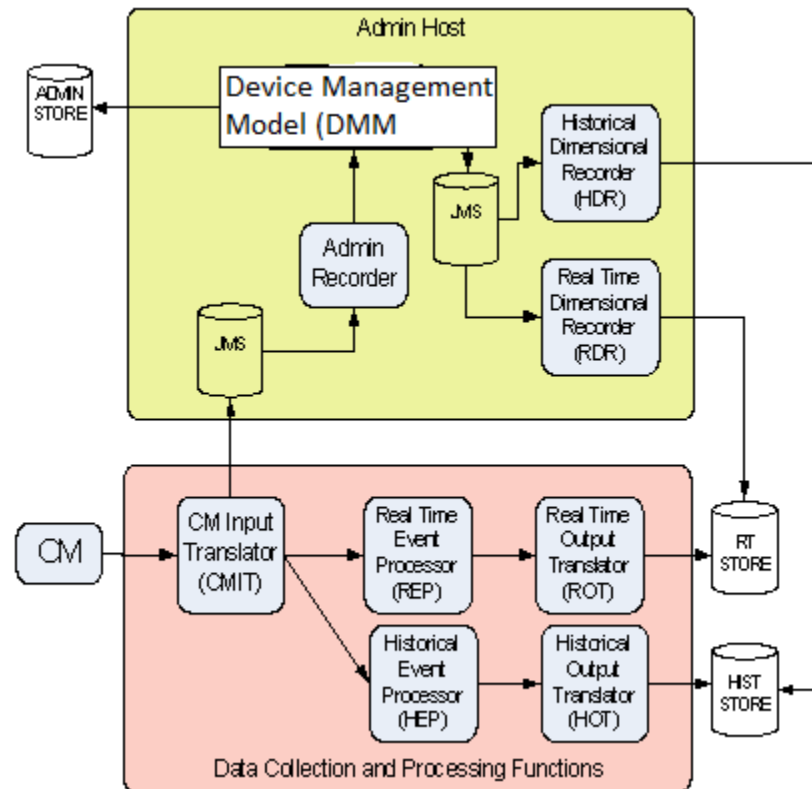
PE RDR Entity Monitor

This process monitors the RDR. The Real Time Dimensional Recorder (RDR) processes administrative data and names collected from one or more external data sources and record it in the real time database.

Container: Admin JBoss.

If this PE fails: Real time administration data is not recorded. Real time administration data includes initial synchronization information and updates to names of agents, queues, routing points, and so on.

Overview of the Processes and Data flow





The Power of We™

1. CMIT receives administration data (Admin pump up) messages and delivers them to the JMS queue on the Admin host, then CMIT requests operational data (OPER pump-up).
2. Admin Recorder processes administration messages from the JMS queue once they arrive, sending them to DMM (via DSS) which delivers them to JMS queues to which HDR and RDR are subscribed.
3. Once operational data is being received, CMIT forwards the messages to HEP/REP. This can occur while Admin Recorder continues to process administration data from the JMS queue.
4. OPER pump-up completes and CMIT receives NAMES pump-up and delivers all names messages to the JMS queue on the Admin host. Admin Recorder processes name messages from the JMS queue sending them to DMM (via DSS) which delivers them to JMS queues to which HDR and RDR are subscribed.

NOTE: Admin data flows to RDR -> Real-time Store and to HDR -> Historical Store, while operational and contact data flows to REP → ROT → Real-time Store and HEP → HOT → Historical Store.

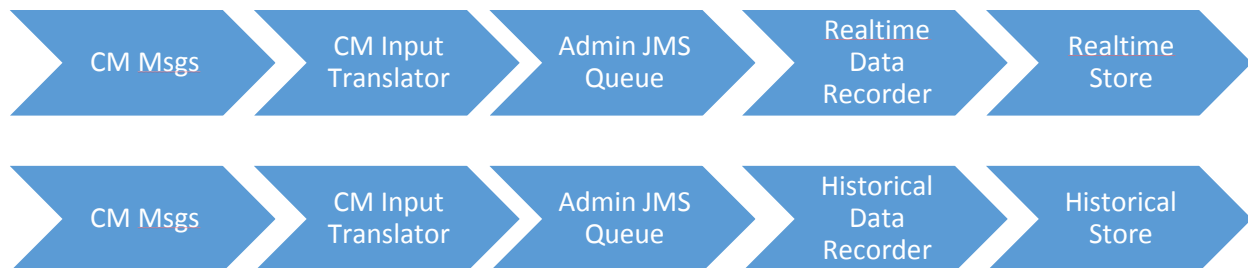
General Issues

Below is the issue which is reported by customers relating to the CM resource population and synchronization between Communication Manager and Avaya IQ.

- Agent has been added in the CM, but the same is not visible in IQ.
 - The entity might differ, but the core issue would remain the same. The troubleshooting steps might differ only in terms of the entity.
 - As known, the entity can be agent, queue, routing point, device and device groups.

Troubleshooting Steps

- When the pump-up begins, the data related to the entity first arrives to the CMIT, hence the logs are first written to the hex_dump_all.log and messages_cmit.log, on the Data collector Host/DataProcessing Host/All in one host for the related DP container.
- From the CMIT the data flows through the JMS queue to the RDR and HDR, thus we need to check the messages_rdr.log and messages_hdr.log files in the REPORTING_RECORDERS folder on the Reporting Host/ All in one host.
- From the Real-time Data Recorder and Historical Data Recorder the data moves into the Database to Real-time and Historical Stores.





The Power of We™

In the below mentioned example, the customer reported that the agent named “David Moyes” was created in the CM whose agent id is “20299”. We will trace the messages through the different log files as these messages pass through the different processing elements that are mentioned in the document before they are written into the database. All the log files are located under the /var/log/Avaya/CCR directory;

Directory	Log file name	Details
DataProcessingJBoss_SOURCENAME/* Or DataCollectionJBoss_SOURCENAME/*	DataProcessingJBoss.log_SOURCENAME.log Or DataCollectionJBoss.log_SOURCENAME.log	Contains logged events from all components in the Data Processing JBoss container.
	messages_cmmsgs.log	Contains all messages sent to and received by the CM Message Parser. The CM Message Parser decodes Communication Manager events from their original binary hexadecimal format into a format suitable for the CMIT.
	messages_cmit.log	Contains all messages sent to and received by the CMIT. The CMIT normalizes event content coming from Communication Manager before sending it to the Avaya IQ historical and real-time event processors. Normalizing content can include performing calculations, filling in missing fields, and so on.
	hex_dump_all.log	Contains all hexadecimal messages received from Communication Manager. Currently, there is no hex_dump_all.log file for Proactive Contact. Use the pcit and pcmsgs logs to monitor synchronization.

Step 1 – CM Adapter Messages

[illegible]

How to read the above message:

1. If we are able to identify the name of the agent or the agent id in the output message, then we can be sure that the CMAAdapter in IQ has received the message for the newly added agent from CM.
2. If we are unable to find the newly added agent details in the messages_cmmsgs.log, this would mean that the CM has not notified IQ of the agent creation and hence, the agent details have not flowed into IQ.

messages_cmit.log:

```
2013-01-31 14:57:19,699 [99.INPUTTRANSLATOR] DEBUG com.avaya.ccr.common.adapter.InputTranslator : 99 :  
INPUT :  
ADMNAMAGENT20:[CMPumpup=int(1),CMTime=int(1)];[End=int(0),MessageId=String(99:1359141144492),Name  
1Flag=int(1),Name2Flag=int(0),Name2ScriptTag=int(0),NameAscii=String(David  
Moyes),NameUTF=String(0),Position=String(20299),Reason=int(1),ReportingSourceUUID=String(079b30673a40fb3  
e013a40fb454b0001),SourceUUID=String(079b30673c72e5c7013c72ed36770002),TimeTag=timestamp(13596211  
06000)]
```

```
2013-01-31 14:57:19,767 [99.INPUTTRANSLATOR] DEBUG com.avaya.ccr.common.adapter.InputTranslator : 99 :  
OUTPUT :  
HDRAgentNames:[HDR=int(1)];[AgentID=String(20299),EvtTimestamp=timestamp(1359621106000),NameASCII=St  
ring(David Moyes),NameUTF=String(David  
Moyes),PartyAccountEID=int(8037),PartyAccountHint=long(1359624439767),PartyEID=int(8037),PartyHint=long(1  
359624439750),SourceEID=int(2504),SourceType=String(ns.cm),SourceUUID=String(079b30673c72e5c7013c72ed3  
6770002),wkmttype=String(ADMIN)]
```

How to read the above message:

1. We now see that CMAAdapter has translated the message from the HEX that it received from CM and the CM Input Translator has received the message (from CMAAdapter) and it shows as INPUT. The CMIT will format the message and pass the resulting OUTPUT messages to the JMS Queue->AdminRecorder->DSS->DMM->HDR/RDR for further processing. We can identify that this is an Administration message by looking at the "wkmttype" parameter which states "ADMIN".

messages_rdr.log:

```
2013-01-31 14:57:21,251 [ActiveMQ Session Task] DEBUG com.avaya.ccr.common.servicecomponent.ee.EESender  
: INPUT :  
HDRAgentNames:[RDR=String(079b30673c72e5c7013c72ed36770002)];[AgentID=String(20299),ChangeInd=int(1),  
EvtTimestamp=timestamp(1359621106000),NameASCII=String(David Moyes),NameUTF=String(David  
Moyes),PartyAccountEID=int(8037),PartyAccountHint=long(1359624439767),PartyEID=int(8037),PartyHint=long(1  
359624439750),SourceEID=int(2504),SourceName=String(null),SourceType=String(ns.cm),SourceUUID=String(079b  
30673c72e5c7013c72ed36770002)]
```

```
2013-01-31 14:57:21,588 [ActiveMQ Session Task] DEBUG com.avaya.ccr.common.servicecomponent.ee.EESender  
: INPUT : CREATE:[RDR=String(079b30673c72e5c7013c72ed36770002)];[name=String(David  
Moyes),parentId=String(079b30673c72e5c7013c72ed486300a6),parentName=String(DADS:  
agents),resourceType=String(agents),sourceId=String(079b30673c72e5c7013c72ed36770002),sourceType=String(c  
m),type=String(resource),uniqueId=String(20299),updateTime=long(1359624441062),updateTimestamp=timestamp  
p(1359624441062)]
```

How to read the above message:

1. Once the RDR received the “HDRAgentNames” message, it will create the same in the Real-time Data Store. This can be identified by the “CREATE” message that we observe above in the messages_rdr.log.
2. If there is no “CREATE” message for the corresponding “HDRAgentNames” message, this would suggest that the data was not inserted correctly in the Real-time data store indicating a possible issue with database/table (In this case table related to the agent entity).

messages_hdr.log:

2013-01-31 14:57:21,252 [ActiveMQ Session Task] DEBUG com.avaya.ccr.common.servicecomponent.ee.EESender : **INPUT :**

HDRAgentNames;[HDR=String(079b30673c72e5c7013c72ed36770002)];[AgentID=String(20299),ChangeInd=int(1),EvtTimestamp=timestamp(1359621106000),NameASCII=String(**David Moyes**),NameUTF=String(David Moyes),PartyAccountEID=int(8037),PartyAccountHint=long(1359624439767),PartyEID=int(8037),PartyHint=long(1359624439750),SourceEID=int(2504),SourceName=String(null),SourceType=String(ns.cm),SourceUUID=String(079b30673c72e5c7013c72ed36770002)]

2013-01-31 14:57:21,588 [ActiveMQ Session Task] DEBUG com.avaya.ccr.common.servicecomponent.ee.EESender : **INPUT : CREATE;**[HDR=String(079b30673c72e5c7013c72ed36770002)];[name=String(**David Moyes**),parentId=String(079b30673c72e5c7013c72ed486300a6),parentName=String(DADS: agents),resourceType=String(**agents**),sourceId=String(079b30673c72e5c7013c72ed36770002),sourceType=String(cm),type=String(resource),uniqueId=String(20299),updateTime=long(1359624441062),updateTimestamp=timestamp(1359624441062)]

How to read the above message:

1. Once the HDR received the “HDRAgentNames” message, it will create the same in the Historical Data Store. This can be identified by the “CREATE” message that we observe above in the messages_hdr.log.
2. If there is no “CREATE” message for the corresponding “HDRAgentNames” message, this would suggest that the data was not inserted correctly in the Historical data store indicating a possible issue with database/table (In this case table related to the agent entity).

Troubleshooting and Workarounds

- If you do not find the entity in the DP or DC logs mentioned before in the presentation, that means/points to the entity never coming to IQ. If you identify the entity recorded in the DP/DC logs, but not in RDR/HDR logs, then it may indicate a problem with DSS or JMS communication.
- The next step is to request the customer to perform some modifications on the entity, like change of name while you monitor the logs to see any activity being reported on them.
- If the updated name is not being reflected in HDR/RDR logs, check for DSSExceptions in ADMIN.log and REPORTING_RECORDERS.log as there may be issues with AdminTomcat/DSS preventing messages to be passed from AdminRecorder to DMM. If you see DSSExceptions at the time of the name change activities, try restarting the AdminTomcat container on the Admin/AF host (with customer permission) and have the customer perform the entity change again.
- Even after performing the above steps, if there is no success, try restarting the MessageBrokerService on the Admin Host and the Data Processing/Data Collection Host (with customer permission) and have the customer perform the entity change again.
- Even after restarting the MessageBrokerService, if the logs do not reflect the entity added/modified, then permission is needed from the customer to restart the Data Processing Services for the CM Source.
 - **Note:** This is service affecting, so we need to get the customers permission on the time preference.
- Please monitor the logs discussed before for the entity details that should be populated in IQ during the CM Translation pump-up.
- Even after restarting the Data Processing Services, if the required entities are not populated, we need to clean the DB tables of the current entries and try a complete fresh pump-up.
 - **Note:** To perform the clean-up and initiate a new pump up, it is recommended to engage the Avaya Backbone/Services to assist as this impacts the database tables, and any mistake can cause cascading issues with the IQ application.