

# RDAQ – Remote Data Acquisition



V1.2 – February 2<sup>nd</sup>, 2014

RDAQ is series of operations centered around the CIQRDAQ.asmx web service. The web service is designed to retrieve data from remote Control IQ sites.

## Web Service

Provide the Web Service with the following variables through a structure.

- Property ID – This is unique to each property
- RequestType – This is constant for the method to return the data
- Request – A set of instruction that will retrieve data from the remote site

Response from the Web Service

- MsgID – This is a reference used for debugging
- dbTime – The date and time of the remote database
- Response – The response base on the RequestType and Request

Requests and responses are processed by the RDAQ (Remote\_Data) driver in CIQProc (requires v4.0 or higher).

## Property ID

This is the property ID of the remote site. This is provide by SAI. It will be an integer

## RequestType

The data will be returned in the format that is provide on the web service request. Options are:

- ~~CSV – Comma Separate Variable with CR/LF between rows~~
- ~~TAB – Tab Separate Variable with CR/LF between rows~~
- XML – XML in table format
- ~~SCHEMA – The Schema of the table in XML format~~
- JSON – Data in JSON format (column names include)
- JSONLite – Data in JSON format (columns names omitted)

Items crossed out may still return data, but there functions have been replaced.

Note: There is a maximum number of rows that will be returned in any table. This is determined by a Preferences tables Key called MaxMobileRows. The default value for this is 100.

## Request

**RDAQ,request**,variable\_1,variable\_2,variable\_n

Where:

RDAQ is a constant. This must be the first entry on each request  
request is one of the following:

<b>Start</b>	- Start the process and wake up the remote Proc
<b>Login</b>	- Login into the remote Proc
<b>Groups</b>	- Get a list of Groups
<b>Views</b>	- Get a list of Views
<b>Points</b>	- Gives a list of points (in a group)
<b>Details</b>	- Lists details of a specific point (alias)
<b>Status</b>	- Displays system status variables
<b>ChangeOptions</b>	- Gets a list of options to change a point's value
<b>ChangeValue</b>	- Parameter to affect a change to a point's value
<b>WhoAml</b>	- Returns the username, viewmask and editmask
<b>End</b>	- Puts the remote Proc in sleep mode
<b>RunHours</b>	- Collects Runhours for Synergy
<b>Header</b>	- Returns just the header data (alarm updates)
<b>AH</b>	
<b>PMS</b>	

Entries are separated by commas (,)

No part of the RDAQ request is case sensitive.

If text is **blue**, these are constants and must be provide

If text is in *italics* you must supply a variable in place of the text

If text is surround by bracket [,variable] it is optional

If more than one entry per command is optional, you must put it in the correct position.

For example, if you want to allow option 2, but not option 1 you would have entry like

RDAQ,REQUEST,useruid,,option\_2

If you don't want either option 1 or 2 you can leave out the commas

RDAQ,REQUEST,useruid

### NOTES:

- ALL requests except for the 'START' request will return a 'Header' table with the database time (dbTime) and the icon name to display for alarms (AlarmFile)
- ALL requests except for the 'START' and 'LOGIN' request must include the encrypted UserID that is returned by the 'LOGIN' command. The userID expires each day at midnight.

## **RDAQ,START**

You should always issue the Start command first. This wakes up the remote proc and starts decrementing the sleep timer.

There are no options on the Start command.

If the property ID is valid, you will get back a table with the site name

If the property ID is '-1' you will get back a table of enabled site names and propID

If the property ID is not valid you will get back a message

Tables Returned: Result

See Also: RDAQ,End

## **RDAQ,LOGIN,name,password**

Name and password are defined in the Control IQ database. This will return the 'Header' table (described above) as well as a 'Login' table that will have a single entry. If the name / password is valid, a 24 character 'key' will be returned that is the encrypted UserID. This key must be included on all other requests (except START).

If the name / password is not valid you will get back a message in the UserID field

"ERROR - Invalid Login or Password".

Tables Returned: Header, Login

See Also: RDAQ,WhoAml

## **RDAQ,GROUPS,useruid,[LastModifiedDate],[Filter]**

UserID comes from the Login request and is required. LastModDate is an optional field to test if the data has changed. If ANY of the data has changed, the full result set will be returned. If a Filter is provided, only the Group Names that contain the filter will be returned.

This will return the 'Header' table (described above) as well as a 'Groups' table that will have a row for each group returned. The rows will contain the 'GroupUID' and 'Group Name' field. The GroupUID is used to call the POINTS request. If no data is found no table is returned.

Typical use:

RDAQ,GROUPS,userid

Result returns 'Header' with dbtime (2014-02-01 16:23:59)

Result returns 'Groups' with GroupUID and Group Names

Store dbTime locally in myGroupTime

Store the Groups table locally if there in no filter.

RDAQ,GROUPS,userid,myGroupTime

This is used for all subsequent calls. If a group has change you will get a full new list. If not, you will not get a 'Groups' table

Note: Two additional rows are added to this table. These will precede the 'regular' groups.

GroupUID      Group Name

- 1 Alarms
- 2 All Points

Note: If the number of rows exceed the MaxMobileRows setting, only the MaxMobileRows will be returned and a row will be appended to the end of the table that states "Maximum Rows Displayed . . ."

Tables Returned: Header, Groups  
See Also RDAQ,POINTS RDAQ,DETAILS

### **RDAQ,VIEWS,userid,[LastModifiedDate],[Filter]**

UserID comes from the Login request and is required. LastModDate is an optional field to test if the data has changed. If ANY of the data has changed, the full result set will be returned. If a Filter is provided, only the View Names that contain the filter will be returned.

This will return the 'Header' table (described above) as well as a 'Views' table that will have a row for each view returned. The rows will contain the 'ViewUID' and 'View Name' field. The ViewUID is used to call the VIEWDETAILS request (*future version*). If no data is found no table is returned.

Typical use:

```
RDAQ,VIEWS,userid
Result returns 'Header' with dbtime (2014-02-01 16:23:59)
Result returns 'Views' with ViewUID and View Names
Store dbTime locally in myViewsTime
Store the Views table locally if there is no filter
```

```
RDAQ,VIEWS,userid,myViewTime
```

This is used for all subsequent calls. If a group has change you will get a full new list. If not, you will not get a 'Views' table

Note: If the number of rows exceed the MaxMobileRows setting, only the MaxMobileRows will be returned and a row will be appended to the end of the table that states "Maximum Rows Displayed . . ."

Tables Returned: Header, Views

### **RDAQ,POINTS,userid,GroupUID,[LastModDate],[Filter]**

UserID comes from the Login request and is required.

GroupUID comes from the Groups request and is required.

LastModDate is an optional field to test if the data has changed. If ANY of the data has changed, the only the changed rows will be returned. If a Filter is provided, only the Points that contain the filter will be returned.

Typical use:

RDAQ,POINTS,userid,groupid

Result returns 'Header' with dbtime (2014-02-01 16:23:59)

Result returns 'Points' a series of fields in multiple rows.

Store dbTime locally in myPointsTime

RDAQ,POINTS,userid,groupid,myPointsTime

This is used for all subsequent calls. If a point has change you only that row returned. If not, you will not get a 'Points' table

Note: If the number of rows exceed the MaxMobileRows setting, only the MaxMobileRows will be returned and a row will be appended to the end of the table that states "Maximum Rows Displayed . . ."

Tables Returned: Header, Points, PointName

See Also: RDAQ,GROUPS

### **RDAQ,DETAILS,userid,alias**

UserID comes from the Login request and is required.

Alias come from the Points table (double-click options) "Details" or is typed in on the details page.

This will return a the 'Header' table (described above)

In addition it will return a 'Details' table with a number of rows that contain 'key' and 'value' fields. The Name is returned in a separate table called 'Point Name'

Tables Returned: Header, Details

See Also: RDAQ,POINTS

### **RDAQ,Header,userid**

UserID comes from the Login request and is required.

DataStream is a text stream either CSV, JSON, JSONLite, XML that defines a dataset

Tables Returned: Header

### **RDAQ,ChangeOptions,userid,alias**

UserID comes from the Login request and is required.

Alias come from the Points table (double-click options)

This will return a the 'Header' table (described above)

In addition, this will return two other tables ('ChangeTo' and 'Actions'). These are to be displayed in the appropriate drop downs in the order they come back. This request will be invoked from a double-click option on the points table "Change Value"

Tables Returned: Header, ChangeTo, Actions

### **RDAQ,ChangeValue,userid,alias,value,action**

UserID comes from the Login request and is required.

Alias come from the Points/Change Form

Value comes from the Points/Change ('Change To' dropdown) Note some values will be returned as the Index of the dropdown (Zero base) or the actual Value in the dropdown as determined by the "I" or "V" in the 'ChangeTo' table.

Action comes from the Points/Change ('Action' dropdown). Return the index (0 based)

Tables Returned: Header

This is the only RDAQ request that changes any data in Control IQ.

### **RDAQ,STATUS,userid**

UserID comes from the Login request and is required.

This will return a the 'Header' table (described above)

In addition it will return a 'Status' tables with a number of rows that contain 'key' and 'value' fields.

Tables Returned: Header, Status

### **RDAQ,WhoAml,userid**

UserID comes from the Login request and is required.

This will return the 'Header' table (described above) as well as a 'WhoAml' table that will the following field and data: UserName, UserID (in Plain text), ViewMask, EditMask

Tables Returned: Header, WhoAml

See Also: RDAQ, Login

### **RDAQ,END,userid**

UserID comes from the Login request and is required.

This collects Runhours from the remote Proc based on a table (XML) stream with a single field (multiple rows) containing the Alias)

This will return the 'Header' table (described above)

This will also return the 'RunHours' table containing Alias (string) and RunHours(Float)

Tables Returned: Header, Runhours

See Also: RDAQ,START

### **RDAQ,RunHours,userid,method,datastream**

UserID comes from the Login request and is required.

Method is either "CSV" or "XML" this is required

DataStream is a text stream either CSV or XML that defines a dataset

This instructs the remote Proc that we are done processing data. This command is usually is not needed (and should not be sent) in that the remote site will time-out on its own after 5 minutes of inactivity. This is primarily used for debugging.

This will return the 'Header' table (described above)

Tables Returned: Header

See Also: RDAQ,START

### **RDAQ,DataEntrySetup,userid**

Tables Returned: Header, Groups, Points

### **RDAQ,DataEntrySubmit,alias,value,timestamp,notes[,alias,value,timestamp,notes][,alias,value,timestamp,notes...]**

```
-1 MSG comes from Program Error comes in the Msg Variable
-2 Msg = "Invalid or Expired UserID"
0  Msg = "Good Data"
1  Msg = "Invalid Username or Password"
2  Msg = "Missing Username or Password"
```

```
3      Msg = "No data matches the pattern"
4      Msg = "Missing UserID"
5      Msg = "Invalid Request"
6      Msg comes from the call Details - NAK
7      Msg comes from the call Details - Alias not Found
8      Msg = "Missing UserID or Alias is too short (3 characters minimum)"
9      Msg = "Missing UserID, Alias or User is not allowed to edit this"
10     Msg = "Missing UserID or GroupUID"
11     Msg comes from the call ChangeValue - NAK
12     Msg = "Alias not found"
13     Msg = "Missing UserID, Alias, Value or Action"
14
15
16
17
18
19
20
21
22
```



## Typical Process

App wakes up

If propertyID / username / userid / SiteName is stored then

RDAQ,START

Show Flash Screen for 5 seconds

RDAQ,Login

Busy icon until you get a return

Present Login Screen if userid invalid - RDAQ,Login

Busy icon

RDAQ,GROUPS,userid,myGroupsDate

RDAQ,VIEWS,userid,myViewsDate (future version)

Store groups / myGroupsDate as necessary

Store views / myViewsDate as necessary

Stop busy icon

Else

Present Login Screen

Busy icon

RDAQ,Login

RDAQ,Start

Store PropID, Login, userid, Sitename,

RDAQ,GROUPS,userid

RDAQ,VIEWS,userid (future version)

Store groups, myGroupsDate

Store views, myViewsDate

Stop busy icon

End IF

Present the CIQ Mobile Home Page