SGCO Dynamic RMS / RDB API For OpenVMS

Stirling&Young

Let's get technical

We appreciate that every installation is different and as a result, most administrators/CIOs will have questions that may be unique to their network. If you don't see the answers you need below feel free to submit your questions directly to our software engineers. You will receive a personal email with the answers to your questions and your submission may be posted on this page as well.

DON'T SEE WHAT YOU NEED TO KNOW?

Submit your questions

What will the SGCO API allow me to do?

Our API gives you a real-time solution to remotely accessing your RMS data not only for reading but also for deletes, updates and writes.

How does the SGCO API work?

Our API will enable a listening port on tcp on your OpenVMS box to all outside access to your RMS data. Our API provides an industry standard REST JSON interface that can be used with all current platforms and programming languages. It has been designed around normal RMS type requests. You send a 4-parameter request that identifies the RMS file name, the simple or compound key number, key value(s), and the action create, remove, update or delete. File locking rules for these actions are determined at the time of installation if defaults are not acceptable.

What is the cost of the SGCP API?

Basic licensing is $7,500 USD per year. $1000.00 USD cluster server

What about installation?

Generally, SGCO will sell a block of hours for the first year to assist with setup, mapping, tuning RMS data files etc. Our API is already working with x86 as well, so any investment if future proofed should you later decide to switch to that platform. Due to the fact that our API uses a single copy of the RMS data rather than making a copy to MsSQL and the declining support from Attunity and Connex we have be receiving many requests.

What does a typical json request looks like ?

Here is a typical API call. This call uses a 3 part compound key with part 2 a partial wildcard and part 3 full wildcard:

{"action": "search", "key_file": "prtlxref", "key": "6", "key_value": "1661|201901||"}

Action will be what you want to do our of CRUD functionality, key file is the RMS file referenced in the config that the API is running as, key is the FDL key number, and key value.

Does the API call involve the path and file name?

The path is defined in the config file depending on the function of that instance. There can be multiple instances of the API running each stated with a command line switch that defines the config section that will apply, This allows instances of the API to run as SSL on port 443 for production, no-SSL on 80 for development etc. as well as to define paths for production and development RMS data files.

Does the API call require the fdl file to be sent or to be read?

FDL files are only needed during initial installation and module creation/mapping.

Can the output from an API call be formatted?

The output values can be formatted however you decide. We often have RMS files that contain binary packed data that needs to be unpacked before delivery.

Here are a couple examples of what is returned from the above query:

RAW JSON:

[{"CONTRACT_COST_WITHOUT_IDC": " 28933.54", "INCOME_START_DATE": "20190102", "SALESMAN_NUMBER": " 4119", "CHILD SALESREP": " ",
"INCOME_STATUS": "A", "VENDOR_NUMBER": " 1661", "CUSTOMER_STATE": "MA", "VENDOR_CUSTOMER_NUMBER": " ", "VENDOR_MASTER": "

", "CALCULATED_MATURITY_DATE": "20240131", "CONTRACT_COST_INCLUDES_IDC": "   29456.60", "SALESMAN_GP": " 9901",
"UP_FRONT_TAX_AMOUNT___TAXES_FIN": "   1933.54", "SALESMAN_P": " 1999", "BOOKING_ACCOUNTING_PERIOD": "201901",
"FIRST_PAYMENT_DUE_DATE": "20190101", "DESCRIPTION_LINE_NO1_PRIMARY": "ELEMENT 2   ", "CUSTOMER_NAME": "SIMPLY P",
"CONTRACT_NUMBER": " 20190001", "CREDIT_LOG_NUMBER": "2018110217"}, {"CONTRACT_COST_WITHOUT_IDC": "   27909.14",
"INCOME_START_DATE": "20190102", "SALESMAN_NUMBER": " 4119", "CHILD SALESREP": "   ", "INCOME_STATUS": "A", "VENDOR_NUMBER": " 1661",
"CUSTOMER_STATE": "NC", "VENDOR_CUSTOMER_NUMBER": "   ", "VENDOR_MASTER": "   ", "CALCULATED_MATURITY_DATE": "20240131",
"CONTRACT_COST_INCLUDES_IDC": "   28413.70", "SALESMAN_GP": " 9901", "UP_FRONT_TAX_AMOUNT___TAXES_FIN": "   0.00", "SALESMAN_P": "
1999", "BOOKING_ACCOUNTING_PERIOD": "201901", "FIRST_PAYMENT_DUE_DATE": "20190101", "DESCRIPTION_LINE_NO1_PRIMARY": "ELEMENT 2
", "CUSTOMER_NAME": "MATTHEW J.", "CONTRACT_NUMBER": " 20190005", "CREDIT_LOG_NUMBER": "2018120206"},
{"CONTRACT_COST_WITHOUT_IDC": "   29400.00", "INCOME_START_DATE": "20190102",
JSON ARRAY:


```
[
 {
  "CONTRACT_COST_WITHOUT_IDC": "    28933.54",
  "INCOME_START_DATE": "20190102",
  "SALESMAN_NUMBER": "  4119",
  "CHILD SALESREP": "       ",
  "INCOME_STATUS": "A",
  "VENDOR_NUMBER": "  1661",
  "CUSTOMER_STATE": "MA",
  "VENDOR_CUSTOMER_NUMBER": "          ",
  "VENDOR_MASTER": "      ",
  "CALCULATED_MATURITY_DATE": "20240131",
  "CONTRACT_COST_INCLUDES_IDC": "    29456.60",
  "SALESMAN_GP": "  9901",
  "UP_FRONT_TAX_AMOUNT___TAXES_FIN": "     1933.54",
  "SALESMAN_P": "  1999",
  "BOOKING_ACCOUNTING_PERIOD": "201901",
  "FIRST_PAYMENT_DUE_DATE": "20190101",
  "DESCRIPTION_LINE_NO1_PRIMARY": "ELEMENT 2           ",
  "CUSTOMER_NAME": "SIMPLY  P",
  "CONTRACT_NUMBER": "  20190001",
  "CREDIT_LOG_NUMBER": "2018110217"
 },
 {
  "CONTRACT_COST_WITHOUT_IDC": "    27909.14",
  "INCOME_START_DATE": "20190102",
  "SALESMAN_NUMBER": "  4119",
  "CHILD SALESREP": "       ",
  "INCOME_STATUS": "A",
  "VENDOR_NUMBER": "  1661",
  "CUSTOMER_STATE": "NC",
  "VENDOR_CUSTOMER_NUMBER": "          ",
  "VENDOR_MASTER": "      ",
```

```
  "CALCULATED_MATURITY_DATE": "20240131",
  "CONTRACT_COST_INCLUDES_IDC": "    28413.70",
  "SALESMAN_GP": "  9901",
  "UP_FRONT_TAX_AMOUNT___TAXES_FIN": "       0.00",
  "SALESMAN_P": "  1999",
  "BOOKING_ACCOUNTING_PERIOD": "201901",
  "FIRST_PAYMENT_DUE_DATE": "20190101",
  "DESCRIPTION_LINE_NO1_PRIMARY": "ELEMENT 2          ",
  "CUSTOMER_NAME": "MATTHEW ",
  "CONTRACT_NUMBER": "  20190005",
  "CREDIT_LOG_NUMBER": "2018120206"
 },
```


Can decimals be exported as decimals, dates as dates?

Yes. Any format can be delivered.

Can simple calculated exports be done such as (<thickness> / 1000)?

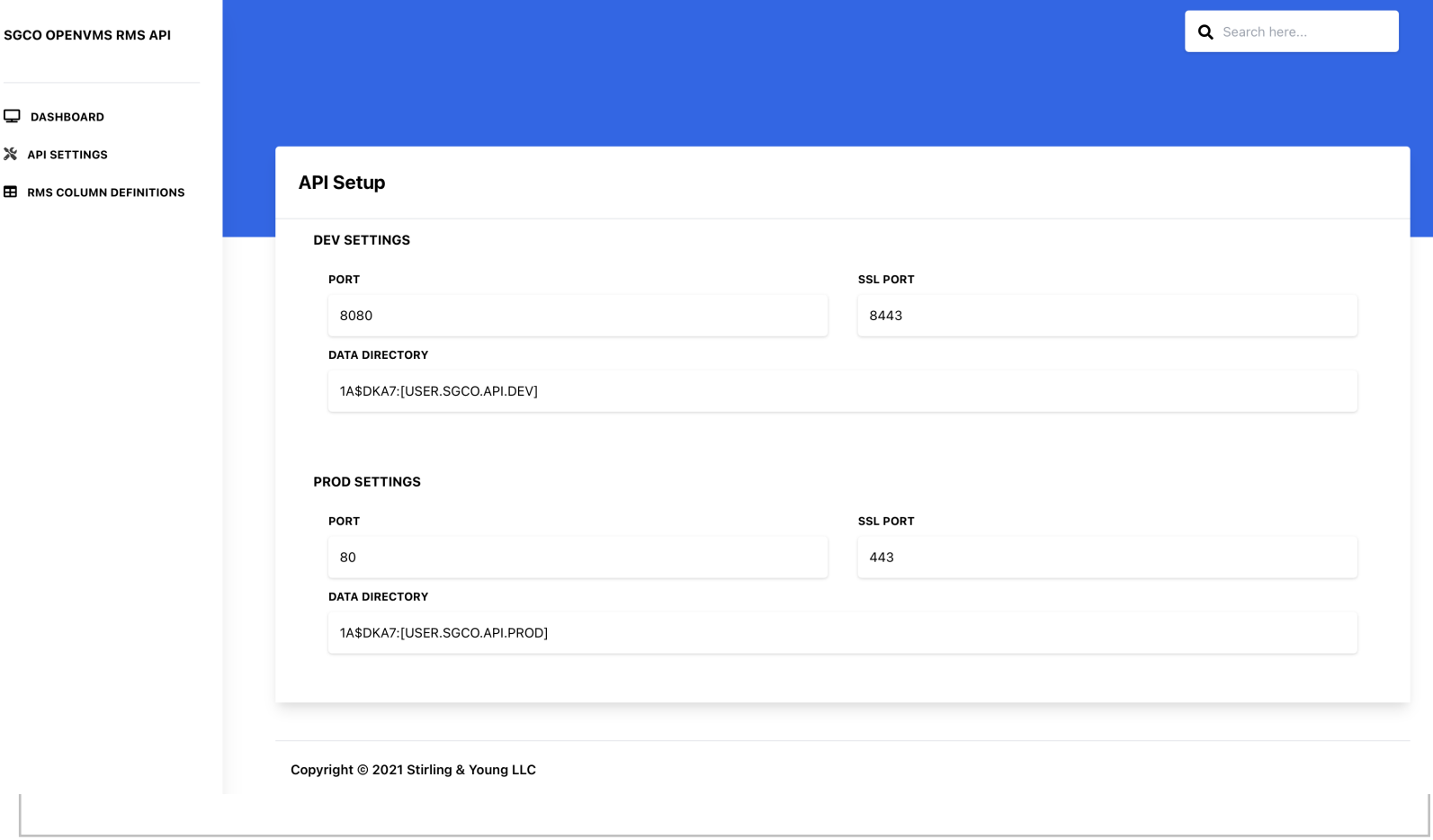
Yes. Pre-processing can to whatever needs to be done before the data leave OpenVMS.

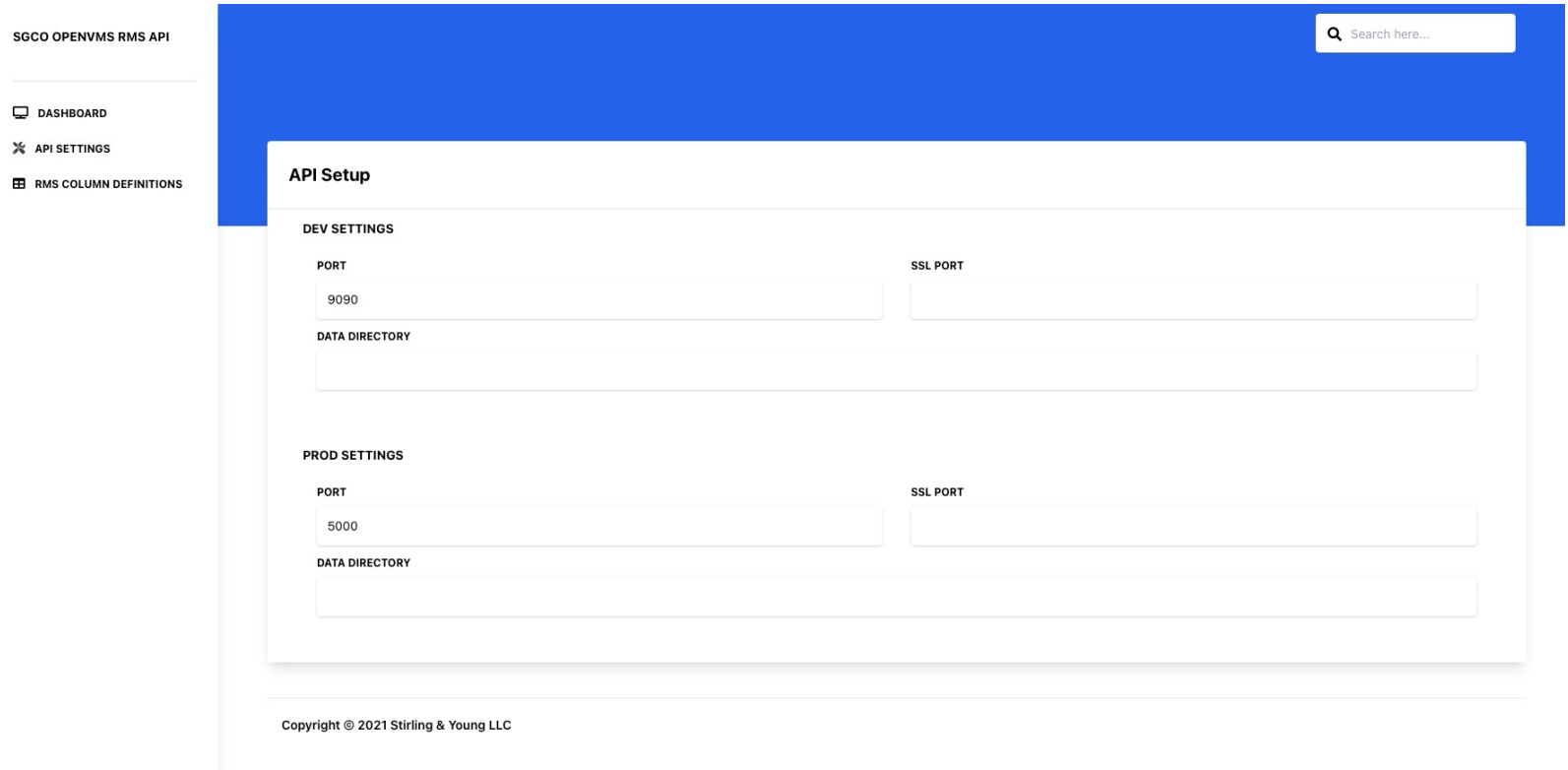
Can a result be a query of multiple files?

Yes, we have the ability to use preprocess logic before returning files. This is much faster to do on OpenVMS side than after delivery to a remote location.

About

Stirling Stirling & Young is an Information Technology consulting firm that delivers innovative, scalable business solutions to help our clients reduce costs, increase revenue and gain competitive advantage through technology. Since 1995 the Stirling Group has provided advice, consultancy, design, implementation, training, network support, internal/external business application solutions and digital media communications solutions to a wide range of companies and organizations in more than 12 countries including USA, Canada, United Kingdom, France, India, Australia, South Africa, Serbia, and the Netherlands.


SGCO OPENVMS RMS API
DASHBOARD
API SETTINGS
RMS COLUMN DEFINITIONS
Search here...
API Setup
DEV SETTINGS
PORT
8080
SSL PORT
8443
DATA DIRECTORY
1A$DKA7:[USER.SGCO.API.DEV]
PROD SETTINGS
PORT
80
SSL PORT
443
DATA DIRECTORY
1A$DKA7:[USER.SGCO.API.PROD]
Copyright © 2021 Stirling & Young LLC


SGCO OPENVMS RMS API
DASHBOARD
API SETTINGS
RMS COLUMN DEFINITIONS
Search here...
API Setup
DEV SETTINGS
PORT
9090
SSL PORT
DATA DIRECTORY
PROD SETTINGS
PORT
5000
SSL PORT
DATA DIRECTORY
Copyright © 2021 Stirling & Young LLC

SGCO OPENVMS RMS API

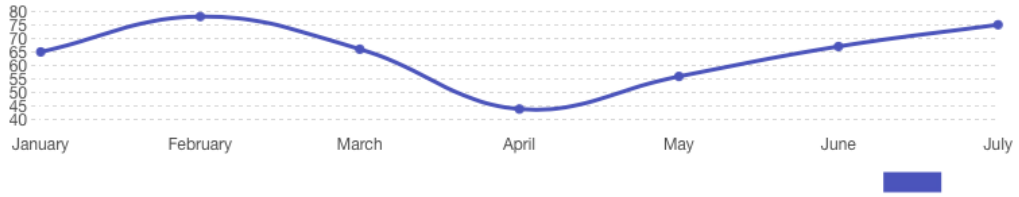
DASHBOARD

API SETTINGS

RMS COLUMN DEFINITIONS

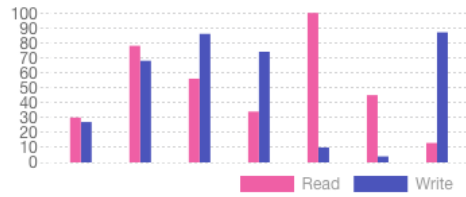
Search here...

API TRAFFIC



PERFORMANCE

Total queries



SGCO OPENVMS RMS API

DASHBOARD

API SETTINGS

RMS COLUMN DEFINITIONS

Search here...

DEMO :: RMS File Definition

PATH

demo.dbs

ALIAS

demo

UPDATE

FIELD DEFINITIONS

NAME	LENGTH	BYTES	ALIGNMENT	AUTO INCREMENT	ACTIONS

ADD FIELD

MAPPING PREVIEW FOR DEMO.DBS

UNMAPPED						1	0	0	0	0							J	o	h	n
UNMAPPED						1	0	0	0	1							t	e	s	t

KEYS

METADATA

Padding byte: ASCII
Filename: demo.dbs

SGCO OPENVMS RMS API

- DASHBOARD
- API SETTINGS
- RMS COLUMN DEFINITIONS

Search here...

DEMO :: RMS File Definition

FIELD DEFINITIONS

NAME	LENGTH	BYTES	ALIGNMENT	AUTO INCREMENT	ACTIONS
ACCESSION_NUMBER	10	0 - 9	< >	☑	DELETE
CUSTOMER_NUMBER	6	10 - 15	< >	☐	DELETE
FIRST_NAME	25	16 - 40	< >	☐	DELETE
LAST_NAME	25	41 - 65	< >	☐	DELETE
EMAIL_ADDRESS	100	66 - 165	< >	☐	DELETE
PASSWORD_HASH	128	166 - 293	< >	☐	DELETE
STATUS_FLAG	1	294 - 294	< >	☐	DELETE
SESSION_ID	120	295 - 414	< >	☐	DELETE

MAPPING PREVIEW FOR DEMO.DBS

ACCESSION_NUMBER	10000
CUSTOMER_NUMBER	
FIRST_NAME	John
LAST_NAME	Young
EMAIL_ADDRESS	jyoung@sgco.com
PASSWORD_HASH	
STATUS_FLAG	
SESSION_ID	
SESSION_DATE	
SESSION_TIME	
NOT_USED	
UNMAPPED	None

ACCESSION_NUMBER	10001
CUSTOMER_NUMBER	
FIRST_NAME	test
LAST_NAME	test
EMAIL_ADDRESS	test@test.com
PASSWORD_HASH	
STATUS_FLAG	
SESSION_ID	
SESSION_DATE	
SESSION_TIME	
NOT_USED	
UNMAPPED	None

KEYS

Key 0: ACCESSION_NUMBER

Key 1: CUSTOMER_NUMBER

Key 2: EMAIL_ADDRESS + ACCESSION_NUMBER

METADATA

Padding byte: ASCII 32 (padded with spaces)
Filename: demo.dbs