

SmartMaster<sup>TM</sup>

2017

---

BTE CORPORATION

ADMINISTRATOR  
MANUAL

## Contents

---

|                                 |    |
|---------------------------------|----|
| Contents .....                  | 1  |
| Introduction.....               | 3  |
| System Functions .....          | 3  |
| Communications.....             | 3  |
| Remote Stations .....           | 3  |
| Applications .....              | 4  |
| System Setup .....              | 4  |
| Software Installation .....     | 5  |
| Server Components .....         | 5  |
| Server Updater.....             | 5  |
| User ID .....                   | 6  |
| Services Update .....           | 7  |
| Viewers Update .....            | 7  |
| Software Updates .....          | 7  |
| Workstation Components .....    | 8  |
| Workstation Updater .....       | 8  |
| User ID .....                   | 8  |
| Viewers Update .....            | 9  |
| Software Updates .....          | 9  |
| Viewers.....                    | 9  |
| Communications .....            | 10 |
| Binding Layer Client .....      | 10 |
| Binding Layer Server .....      | 11 |
| Application Layer Client.....   | 11 |
| Application Layer Server.....   | 12 |
| Remote Stations .....           | 12 |
| Input Managers / Polling .....  | 13 |
| Output Managers .....           | 15 |
| Remote Station Definition ..... | 16 |
| Remote Station Addressing ..... | 16 |
| Comverge RTC Protocol .....     | 17 |
| Telescada ARS Protocol.....     | 17 |
| - Ccu7Cctl.....                 | 17 |
| - TscRTUShortAddr.....          | 17 |
| - TscRTULongAddr.....           | 18 |
| - TscRTURelayAddr.....          | 18 |
| - MotorolaDigital.....          | 18 |
| - GE REMS 101/102 .....         | 18 |
| - SA REMS 101/102 .....         | 19 |
| - SaDigital .....               | 19 |
| - FpDigital.....                | 19 |
| - Sa206 .....                   | 20 |
| TS-11 protocol.....             | 20 |
| I/O Points.....                 | 21 |
| Definition .....                | 21 |
| Configuration .....             | 22 |
| PSP Parameter .....             | 23 |
| Evaluation .....                | 24 |
| Evaluators .....                | 25 |
| - Source Code Authoring.....    | 26 |

|   |    |
|---|----|
| - Source Code Compilation .....                                       | 28 |
| OVM (Output Value Management) .....                                   | 28 |
| Dynamic Operation Control .....                                       | 28 |
| I/O Point Groups .....  | 28 |
| Remote Station Groups .....   | 29 |
| SCADA .....   | 30 |
| Programs .....  | 30 |
| Program Creation .....  | 31 |
| Database Tables .....   | 31 |
| Table Columns .....   | 32 |
| Program Execution .....   | 32 |
| One-Line Diagram Displays .....                                       | 33 |
| Display Designer .....  | 34 |
| Display Element Types .....   | 34 |
| - Atomic Display Elements .....                                       | 34 |
| - Composite Display Elements .....                                    | 34 |
| Display Element Creation .....  | 34 |
| - Line .....  | 35 |
| - Polyline, Polygon .....   | 35 |
| - Circle .....  | 35 |
| - Arc .....   | 36 |
| - Text .....  | 36 |
| - Date / Time Display .....   | 36 |
| - Gauge – Relay_02 (Two-state relay) .....                            | 37 |
| - Gauge – Relay_03 (Three-state relay) .....                          | 37 |
| - Gauge – VBreaker_02 (Two-state breaker, vertical orientation) ..... | 37 |
| - Composite .....   | 38 |
| Change Display Element .....  | 38 |
| - Edit Display Element .....  | 38 |
| - Configure Display Element .....                                     | 39 |
| Change Display Element Children .....                                 | 39 |
| Display Element Colors .....  | 39 |
| - Stroke Colors .....   | 39 |
| - Fill Colors .....   | 39 |
| Mapping of Display Elements .....                                     | 40 |
| Text Display Element Gauging .....                                    | 40 |
| Load Management .....   | 41 |
| Strategies .....  | 42 |
| Demand Inputs .....   | 42 |
| Load Points .....   | 43 |
| Load Groups .....   | 44 |
| Manual Load Point Operation .....                                     | 44 |
| Load Point Operation Priority .....                                   | 45 |
| Secondary Loads .....   | 45 |
| Programs .....  | 46 |
| Demand Response .....   | 46 |
| Load State Reinserting .....  | 47 |
| System Users .....  | 48 |
| System User List .....  | 48 |
| Security Credentials .....  | 49 |
| Notifications .....   | 50 |
| SMWebLink .....   | 52 |

---

## Introduction

---

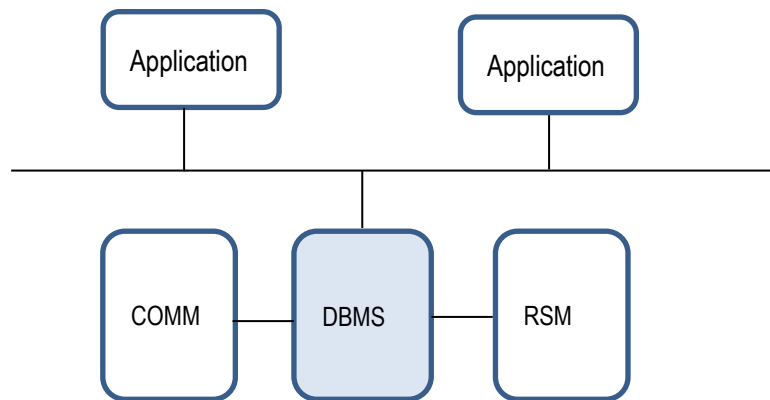
This manual is intended for users who are able to set up and / or reconfigure SmartMaster™ (SM) so it better reflects the monitored distribution system. It is assumed that the reader has read SM User Manual.

---

## System Functions

---

Below is SM block diagram depicting main system functions.



At the highest level, SM performs three functions:

1. Communications (COMM)
2. Management of remote stations (RSM)
3. Applications

---

## Communications

---

The *COMM* block manages communications between SM and remote stations in the field. Its basic function is to get data from remote stations to SM, and to push data from SM to remote stations. In SM, communications are managed by *Communications* service running on the server.

---

## Remote Stations

---

The *RSM* block is Remote Stations Manager. It manages remote stations in the field, such as IEDs, RTUs, switches, etc. It ensures that proper remote station parameters are defined and used, appropriate communications channels are assigned for communications between SM and

respective remote stations, that communications take place at appropriate times, and that all data going to and coming from I/O points (inputs, such as analog and status readings, and outputs such as controls) is properly processed and stored. In SM, these functions are managed by *RemoteStations* service running on the server.

---

## Applications

---

Applications use I/O point database records. Each application also creates its own database records. These applications, such as *SCADA*, *load management*, etc. run algorithms appropriate to those applications, using above mentioned database records. In SM, each application has its own service on the server.

---

## System Setup

---

To set up a functioning SM system, the following need to be set up first in the order listed below:

- Software installation (OS, SQL Server, SM services, SM user interface programs)
- One security account (SQL Server)
- Communications channels that will be used by SM (*Communications* service)
- All remote stations that will be managed by SM need to be defined and properly configured (*RemoteStations* service)
- Input manager(s) (*RemoteStations* service)
- Remote station polling (*RemoteStations* service)
- I/O points need to be defined and properly configured (*RemoteStations* service)
- Evaluators (scripts of source code, authored by the Administrator, which process I/O point data) need to be defined (*RemoteStations* service)

Above constitute a minimum set of objects that need to be defined so SM can operate. Above minimal configuration allows SM to function as a data concentrator or a small, basic SCADA master controller which can poll remote stations for data, and can dispatch control codes.

Beyond these, further system objects need to be defined to get additional functionality. They are listed below in no particular order:

- System backup schedule (*SmartMaster* service)
- Holiday list (*SmartMaster* service)
- Evaluators – for more elaborate data processing logic (*RemoteStations* service)
- Remote station filters (*RemoteStations* service)
- Remote station groups (*RemoteStations* service)
- I/O point groups (*RemoteStations* service)
- Output Managers (*RemoteStations* service)

- Event Log Filters (*Event Viewer*)
- Data Filters (*Data Viewer*)
- System users database (*SystemUsers* service)
- Security accounts (*SystemUsers* service)
- User notifications (*SystemUsers* service and *Database Mail* in SQL Server)
- SMWebLink – web browser access (*SystemUsers* service)
- User Authored Programs (*SCADA* service)
- One-Line Diagrams (*SCADA* service)
- Demand inputs (*LoadManagement* service)
- Load points (*LoadManagement* service)
- Load groups (*LoadManagement* service)
- Demand response (*LoadManagement* service)
- Load state reinserting (*LoadManagement* service)
- Load rotation (*LoadManagement* service)
- Reports (*Report Services* in SQL Server and *Internet Explorer*)
- Auto-failover (SQL Server, SM)

## Software Installation

---

SM system software comes in two parts:

- Server resident components
- Workstation resident components

## Server Components

---

It is assumed that Windows Server OS and SQL Server are installed and properly configured. This is usually done by BTE before server is installed on customer site.

## Server Updater

---

Server Updater is a program which is to execute on the server only. Server Updater installer is not distributed via BTE website for security reasons. The installer is placed in SmartMaster\Updates directory (file SMSUYYYY.1.RRR.MSI) on customer's server.

**STEP 1:** Run Server Updater installer – this installs three programs on the server:

1. User ID – a program which identifies a user associated with specific Windows Account to SM. This program is run one time only so SM knows who is subsequently running the Viewers when logged on to a specific Windows account, and assigns them appropriate security privileges. This program, when running on a server, also configures SM as a primary / secondary system when auto-failover system is being configured.

2. Services Update – a program responsible for updating all SM service components on the server. This program is installed on the server only, and is manually executed when services need to be updated to the latest available version.
3. Viewers Update – a program responsible for updating all Viewers. This program is installed on the server and all workstations running the Viewers. It is manually executed when Viewers need to be updated to the latest available version.

---

## User ID

---

As described above, User ID is used to record user credentials so they do not always have to be entered whenever one of the Viewers (user interface programs), or services are started.

STEP 2: Run User ID, and enter credentials used by SM services to log on to SQL server:

1. Start User ID as Administrator - if not started as administrator User ID is unable to store information in Windows registry.
2. Click on 'Server Remote' six times to un-gray 'Service' check box. The latter is protected with this simple procedure so service credentials are not inadvertently changed which would prevent services from running
3. Make sure 'Services' is checked
4. Make sure 'Server Remote' is unchecked
5. Enter 'Server Name' – this server computer (network) name or IP address
6. Service Instance defaults to SmartMaster – leave unchanged
7. Enter customer Site Code
8. Make sure 'Integrated Security' is unchecked
9. Enter SMSservice for 'User Name'
10. Enter service password
11. Press the Tab key a couple of times until OK button appears
12. Click on OK button
13. Close the program

STEP 3: Run User ID, and enter credentials used by SM Viewer executed under current Windows account. This procedure must be repeated for every Windows account on the server that is used to execute SM viewers.

1. Start User ID as Administrator - if not started as administrator User ID is unable to store information in Windows registry.
2. Make sure 'Server Remote' is unchecked
3. Enter 'Server Name' – this server computer (network) name or IP address
4. Service Instance defaults to SmartMaster – leave unchanged
5. Enter customer Site Code
6. Make sure 'Integrated Security' is checked if current Windows account is to be used to access SQL server, or unchecked if separate user name and password are to be used.
7. Enter 'User Name'
8. Enter password
9. Press the Tab key a couple of times until OK button appears
10. Click on OK button

## 11. Close the program

### Services Update

---

Services Update installs or updates SM services on the server. To install or update SM services proceed as follows:

#### STEP 4: Run Services Update.

1. Start Services Update as Administrator – if not started as administrator Services Update is unable to shut down previously running services, or restart newly installed services.
2. Click on 'Update' button. If currently running version of services is the same as the one found on BTE server, the program asks the user if it is to proceed with installation any way.
3. The program stops and uninstalls previously running services, if any. It then downloads the latest available version from BTE server, installs the downloaded version, and restarts the services in case they were running before installation.
4. If services were not running before installation they must be started manually. All SM services are started / stopped by starting or stopping SmartMaster Service like any other Windows service. For simplicity, all SM services are controlled by this one Windows service.

### Viewers Update

---

Viewers Update installs or updates SM Viewers (UI programs) on the server. To install or update SM Viewers proceed as follows:

#### STEP 5: Run Viewers Update to install or update SM Viewers.

1. Shut down any previously running Viewers on the machine
2. Start Viewers Update as Administrator - if not started as administrator the program is unable to store information in Windows registry.
3. Click on 'Update' button. If currently installed version of viewers is the same as the one found on BTE server, the program asks the user if it is to proceed with installation any way.
4. The program downloads the latest available version from BTE server and installs the downloaded version.

### Software Updates

---

Once installed, updating services to the latest version is a much shorter process. Only Step 4 above needs to be repeated on the server.



Once installed, updating viewers to the latest version is a much shorter process. Only Step 5 above needs to be repeated on the server.

## Workstation Components

---

It is assumed that Windows OS is installed and properly configured.

### Workstation Updater

---

Workstation Updater is a program which is to execute on every workstation which will run SM Viewers. Workstation Updater can be downloaded from BTE website.

#### STEP 1: Download Workstation Updater installer

1. Go to <http://www.btecentral.com>
2. Go to SmartMaster → Downloads
3. Execute the installer (see below), or save the installer in SmartMaster\Updates directory on the workstation hard drive.

#### STEP 2: Run Workstation Updater installer – this installs two programs on the workstation:

1. User ID – a program which identifies a user associated with specific Windows Account to SM. This program is run one time only so SM knows who is subsequently running the Viewers when logged on to a specific Windows user account, and assigns them appropriate security privileges.
2. Viewers Update – a program responsible for updates of all Viewers. This program is installed on the server and all workstations running the Viewers. It is manually executed when Viewers need to be updated to the latest available version.

## User ID

---

As described above, User ID is used to record user credentials so they do not always have to be entered whenever one of the Viewers (user interface programs) is started.

STEP 3: Run User ID and enter credentials used by SM Viewers executed under current Windows account. This procedure must be repeated for every Windows account that is used to execute SM viewers.

1. Start User ID as Administrator - if not started as administrator User ID is unable to store information in Windows registry.
2. Make sure 'Server Remote' is unchecked if User ID is running on a machine connected to a LAN. Make sure 'Server Remote' is checked if workstation will be connecting to the server via the internet.
3. Enter 'Server Name'. This can be server network name if 'Server Remote' is unchecked or server IP address if server is accessed via the internet. Depending on network

configuration, sometimes <IP address>,<Port Number> needs to be entered, where Port Number is the port number used to connect to SQL server, usually 1433. For example, if server address is 1.2.3.4, enter 1.2.3.4,1433. One can also enter AUTO for server IP address – this directs the Viewer to consult BTE server for server IP address.

4. Service Instance defaults to SmartMaster – leave unchanged
5. Enter customer Site Code
6. Make sure 'Integrated Security' is checked if current Windows account is to be used to access SQL server, or unchecked if separate user name and password are to be used. The latter must be used when accessing the server via the internet.
7. Enter 'User Name'
8. Enter password
9. Press the Tab key a couple of times until OK button appears
10. Click on OK button
11. Close the program

---

## Viewers Update

Viewers Update installs or updates SM Viewers (UI programs) on the workstation. To install or update SM Viewers proceed as follows:

**STEP 4:** Run Viewers Update to install or update SM Viewers.

1. Shut down any previously running Viewers on the machine
2. Start Viewers Update as Administrator - if not started as administrator the program is unable to store information in Windows registry.
3. Click on 'Update' button. If currently installed version of services is the same as the one found on BTE server, the program asks the user if it is to proceed with installation any way.
4. The program downloads the latest available version from BTE server and installs the downloaded version.

---

## Software Updates

Once installed, updating the viewers to the latest version is a much shorter process. Only Step 4 above needs to be repeated.

---

## Viewers

Note that only one UI program can run on any one machine at any one time. Each UI program (Viewer) assumes IP address of the host machine to communicate with the server. This is not a problem on workstation machines which are usually PCs. On the server it may become an issue when several people are logged on via remote desktop, for example, each running an instance of UI. In such a case all instances use the same IP address to communicate with the

server and port conflicts occur. So the basic rule to follow is: Only one running UI per machine at any one time.

## Communications

Communications service manages communications between SM and remote stations via communications channels. In SM, each communications channel consists of two parts:

- Binding layer – moves data in and out of the computer
- Application layer – runs communications protocol layers

Communications channels can share the same binding layer. This way SM can run multiple protocols over the same physical channel.

Users can create and delete communications channels as needed. Internal service broker and executive channels should not be deleted as they are used by SM internally for communications between the various services and viewers.

Dynamic screen updating is disabled in communications service window to reduce the overhead on the server.

The screenshot shows the 'Service=Communications' window with the following data:

| Application Layer Clients |         |         |        |     |     |
|---------------------------|---------|---------|--------|-----|-----|
| Name                      | Manager | Running | Status | TXM | RXM |
| _Communications           | Enabled | Yes     | OK     | 0   | 0   |
| _Installations            | Enabled | Yes     | OK     | 0   | 0   |
| _LoadManagement           | Enabled | Yes     | OK     | 0   | 0   |
| _RemoteStations           | Enabled | Yes     | OK     | 0   | 0   |
| _Scada                    | Enabled | Yes     | OK     | 0   | 0   |
| _SMService                | Enabled | Yes     | OK     | 107 | 0   |
| _SystemUsers              | Enabled | Yes     | OK     | 0   | 0   |

| Binding Layer Clients |         |         |           |        |       |     |
|-----------------------|---------|---------|-----------|--------|-------|-----|
| Name                  | Manager | Running | Connected | Status | TXB   | RXB |
| _Communications       | Enabled | Yes     | Yes       | OK     | 1843  | 0   |
| _Installations        | Enabled | Yes     | Yes       | OK     | 534   | 0   |
| _LoadManagement       | Enabled | Yes     | Yes       | OK     | 15030 | 0   |
| _RemoteStations       | Enabled | Yes     | Yes       | OK     | 541   | 0   |
| _Scada                | Enabled | Yes     | Yes       | OK     | 478   | 0   |
| _SMService            | Enabled | Yes     | Yes       | OK     | 29702 | 0   |
| _SystemUsers          | Enabled | Yes     | Yes       | OK     | 0     | 0   |

| Application Layer Servers |         |         |        |     |     |
|---------------------------|---------|---------|--------|-----|-----|
| Name                      | Manager | Running | Status | TXM | RXM |
| Communications            | Enabled | Yes     | OK     | 0   | 3   |
| Installations             | Enabled | Yes     | OK     | 0   | 0   |
| LoadManagement            | Enabled | Yes     | OK     | 0   | 32  |
| RemoteStations            | Enabled | Yes     | OK     | 0   | 0   |
| Scada                     | Enabled | Yes     | OK     | 0   | 0   |
| SMService                 | Enabled | Yes     | OK     | 0   | 298 |
| SystemUsers               | Enabled | Yes     | OK     | 0   | 2   |

| Binding Layer Servers |         |         |           |        |     |        |
|-----------------------|---------|---------|-----------|--------|-----|--------|
| Name                  | Manager | Running | Connected | Status | TXB | RXB    |
| Communications        | Enabled | Yes     | Yes       | OK     | 0   | 411    |
| Installations         | Enabled | Yes     | Yes       | OK     | 0   | 3440   |
| LoadManagement        | Enabled | Yes     | Yes       | OK     | 0   | 16374  |
| RemoteStations        | Enabled | Yes     | Yes       | OK     | 0   | 22504  |
| Scada                 | Enabled | Yes     | Yes       | OK     | 0   | 3372   |
| SMService             | Enabled | Yes     | Yes       | OK     | 0   | 214582 |
| SystemUsers           | Enabled | Yes     | Yes       | OK     | 0   | 1138   |

Events: Connected to \\CANTOR\SmartMaster service, site NYBTE. [OK]

### Binding Layer Client

To define a new binding layer client proceed as follows:

- Start the *Main Viewer*
- Open *Communications* service window
- Right-click anywhere in the *Binding Layer Clients* display
- Execute the *New Client...* command
- Enter the new client name. In SM, client names are preceded with an underscore character ‘\_’.
- From the list of client types select the new client type
- Close and re-open *Communications* service window

The new client appears in *Binding Layer Clients* display. To configure the new client:

- Right-click on the new client
- Execute *Configuration – Client* command
- Modify parameters as needed. At the very least one must set the value the new object *Manager* to *Enabled*.
- Select *Save + Exit*

## Binding Layer Server

---

To define a new binding layer server proceed as follows:

- Start the *Main Viewer*
- Open *Communications* service window
- Right-click anywhere in the *Binding Layer Servers* display
- Execute the *New Server...* command
- Enter the new server name.
- From the list of server types select the new server type
- Close and re-open *Communications* service window

The new server appears in *Binding Layer Servers* display. To configure the new server:

- Right-click on the new server
- Execute *Configuration – Server* command
- Modify parameters as needed. At the very least one must set the value the new object *Manager* to *Enabled*.
- Select *Save + Exit*

## Application Layer Client

---

To define a new application layer client proceed as follows:

- Start the *Main Viewer*
- Open *Communications* service window

- Right-click anywhere in the *Application Layer Clients* display
- Execute the *New Client...* command
- Enter the new client name. In SM, client names are preceded with an underscore character ‘\_’.
- From the list of client types select the new client type
- From the list of binding layer clients select a client
- Close and re-open *Communications* service window

The new client appears in *Application Layer Clients* display. The last step above associates application layer client with a binding layer counterpart. Associating clients with servers is not supported. To configure the new client:

- Right-click on the new client
- Execute *Configuration – Client* command
- Modify parameters as needed. At the very least one must set the value the new object *Manager* to *Enabled*.
- Select *Save + Exit*

## Application Layer Server

---

To define a new application layer server proceed as follows:

- Start the *Main Viewer*
- Open *Communications* service window
- Right-click anywhere in the *Application Layer Servers* display
- Execute the *New Server...* command
- Enter the new server name.
- From the list of server types select the new server type
- From the list of binding layer servers select a server
- Close and re-open *Communications* service window

The new server appears in *Application Layer Servers* display. The last step above associates application layer server with a binding layer counterpart. Associating servers with clients is not supported. To configure the new server:

- Right-click on the new server
- Execute *Configuration – Server* command
- Modify parameters as needed. At the very least one must set the value the new object *Manager* to *Enabled*.
- Select *Save + Exit*

## Remote Stations

---

SM sends data to remote stations, and receives data from remote stations. The latter is coordinated by *input managers* in RemoteStations service.

## Input Managers / Polling

---

Polling remote stations for data can be manually initiated by an operator, or periodic. At the end of every poll cycle, after all data has arrived from remote stations, evaluation cycle executes. This cycle performs all calculations on the newly arrived data by executing evaluators defined by system administrator. It is important that evaluation cycle happens last, i.e. after polling cycle is complete.

*Input managers* execute periodic remote station polls. In addition, if so configured, a designated input manager is also a trigger for evaluation cycles. There should one, and only one input manager designated as an Evaluation Cycle Trigger (ECT). If multiple input managers are defined, the one which executes (or whose execution ends) last should be designated as an ECT.

To define a new input manager:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click anywhere in the *Input Managers* display
- Execute *New Input Manager...* command
- Enter new input manager's *name*

The new input manager appears in the *Input Managers* display.

Service=RemoteStations

**Remote Stations** 11:06 35

**I/O Point Groups**

| Name         | Manager | Status |
|--------------|---------|--------|
| PowerPlant   | Enabled | OK     |
| SESubstation | Enabled | OK     |

**Remote Station Groups**

| Name         | Manager | Status |
|--------------|---------|--------|
| PowerPlant   | Enabled | OK     |
| SESubstation | Enabled | OK     |

**Input Managers**

| Name         | Manager | Running | Status | LastPollDT          | NextPollDT          |
|--------------|---------|---------|--------|---------------------|---------------------|
| PowerPlant   | Enabled | Yes     | OK     | 2013-09-28 11:06:00 | 2013-09-28 11:07:00 |
| PowerPlantM  | Enabled | Yes     | OK     | 2013-09-28 11:06:00 | 2013-09-28 11:07:00 |
| SESubstation | Enabled | Yes     | OK     | 2013-09-28 11:06:00 | 2013-09-28 11:07:00 |

**Output Managers**

| Name         | Manager | Running | Status | LastExecutionDT     | NextExecutionDT     |
|--------------|---------|---------|--------|---------------------|---------------------|
| PowerPlant   | Enabled | Yes     | OK     | 2013-09-28 11:06:20 | 2013-09-28 11:07:20 |
| PowerPlantM  | Enabled | Yes     | OK     | 0001-01-01 00:00:00 | NONE                |
| SESubstation | Enabled | Yes     | OK     | 0001-01-01 00:00:00 | NONE                |

**Filters**

| Name         |
|--------------|
| All          |
| PowerPlant   |
| SESubstation |

**Remote Stations**

| Name            | Manager | Status | MRT0 | NRC | InputManager |
|-----------------|---------|--------|------|-----|--------------|
| Brownville      | Enabled | OK     | 670  | 0   | SESubstation |
| EastFeeder      | Enabled | OK     | 670  | 0   | SESubstation |
| Feeder3         | Enabled | OK     | 639  | 0   | PowerPlant   |
| Feeder4Recloser | Enabled | OK     | 546  | 0   | PowerPlant   |
| Feeder5         | Enabled | OK     | 686  | 0   | PowerPlant   |
| Feeder6Recloser | Enabled | OK     | 546  | 0   | PowerPlant   |
| GEN1            | Enabled | OK     | 546  | 0   | PowerPlant   |
| GEN2            | Enabled | OK     | 530  | 0   | PowerPlant   |
| GEN4            | Enabled | OK     | 468  | 0   | PowerPlant   |
| GEN5            | Enabled | OK     | 561  | 0   | PowerPlant   |
| GEN6            | Enabled | OK     | 546  | 0   | PowerPlant   |
| GEN7            | Enabled | OK     | 468  | 0   | PowerPlant   |
| Howe            | Enabled | OK     | 670  | 0   | SESubstation |
| JohnsonFeeder   | Enabled | OK     | 655  | 0   | PowerPlant   |

Events Window RS\_DNP\_30=GEN6 was opened. OK

To configure the new manager:

- Right-click on the new object
- Execute *Configuration* command
- Modify parameters as needed. At the very least one must set
  - The value the new object *Manager* to *Enabled*

- *CommChannel* parameter - this associates one of the application layer clients with the new input manager
- The *ECT* parameter designates the new input manager as an Evaluation Cycle Trigger (see discussion above).
- Select *Save + Exit*

## Output Managers

---

Output managers run in RemoteStations service. They perform several functions, each of which can be enabled or disabled:

- Output Value Management (OVM) – This is enforcement of computed output values. User written SCADA programs compute output values of the various outputs. Output managers send control codes to the various outputs, making sure that output states in the field are in congruence with their respective output values in the database.
- Timed Switch Management (TSM) – This is updating of timed switch states before they time out.
- Remote Station Time Synchronization (RSTS) – This is dispatching of periodic time synchronization messages to remote stations that consume such messages.
- Notifications cycle trigger – This is triggering of notification dispatcher, executes every time output manager executes (no configuration parameters)

To define a new output manager:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click anywhere in the *Output Managers* display
- Execute *New Output Manager...* command
- Enter new output manager's *name*

The new output manager appears in the *Output Managers* display.

To configure the new manager:

- Right-click on the new object
- Execute *Configuration* command
- Modify parameters as needed. At the very least one must set
  - The value the new object *Manager* to *Enabled*
  - *CommChannel* parameter - this associates one of the application layer clients with the new output manager
  - *ExecutionPeriod* must be set to a value greater than zero, otherwise output manager never executes. Execution period is in seconds.
  - *OVM* must be *Enabled* if output value management is to execute.
  - *TSM* must be *Enabled* if time switch management is to execute.
  - *RSTSPeriod* must be set to a value greater than zero for time synchronization of remote stations to execute.



- Select *Save + Exit*

---

## Remote Station Definition

---

*RemoteStation* managers run all activities associated with remote station management. To define new *RemoteStation* object:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click anywhere in the *Remote Stations* display
- Execute *New Remote Station...* command
- Enter new remote station *name*
- Select remote station *type* from the *Station Type* list

Alternatively, new remote station can be created from an existing one by executing the *Copy Remote Station...* command:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click on to remote station to be copied
- Execute *Copy Remote Station...* command
- Enter new remote station *name*

Note that copying remote station copies all remote station parameters, as well as all associated I/O points. At the very least the new remote station address needs to be changed.

The new remote station appears in the *Remote Stations* display. To configure remote station:

- Right-click on the new object
- Execute *Configuration* command
- Modify parameters as needed. At the very least one must set
  - The value the new object *Manager* to *Enabled*.
  - *InputManager* – parameter determines input channel and scheduling
  - *OutputManager* – parameter defines output channel and scheduling
  - *Address* – remote station address, to distinguish it from all others. Format and number of addressing parameters varies depending on remote station type (see below)
- Select *Save + Exit*

---

## Remote Station Addressing

---

Remote station address parameters are self-documented, as described in the *User Manual*. Nevertheless, since some protocols have a variety of addressing options, they are documented here.

## Protocols

| PROTOCOL            | ADDR. PARAMETER   | COMMENTS                        |
|---------------------|-------------------|---------------------------------|
| <b>Comverge RTC</b> | SerialNumber      | Used for programming the switch |
| <b>Comverge RTC</b> | OpAddress0        | Switch operational address #0   |
| <b>Comverge RTC</b> | OpAddress1        | Switch operational address #1   |
| <b>Comverge RTC</b> | OpAddress2        | Switch operational address #2   |
| <b>Comverge RTC</b> | OpAddress3        | Switch operational address #3   |
| <b>Comverge RTC</b> | OpAddress4        | Switch operational address #4   |
| <b>Comverge RTC</b> | OpAddress5        | Switch operational address #5   |
| <b>DNP 3.0</b>      | Address           | Destination Address, 16 bits    |
| <b>Modbus</b>       | Address           | Modbus Address, 8 bits          |
| <b>TS-11</b>        | Address           | Switch Address (0 – 511)        |
| <b>NPPD</b>         | GroupDesignations | Groups A - K                    |

### Comverge RTC Protocol

SM support of Comverge RTC protocol is there mainly to support operation of Comverge 205 switches. At this time it is also possible to operate SA Digital switches by setting up Comverge 205 driver parameters as follows:

- Use Operational Address # 0 for switch address
- Use State0FunctionCode1 parameter to enter *Mark Tone* value
- Use State0FunctionCode2 parameter to enter *Space Tone* value
- Use State1FunctionCode1 parameter to enter *Mark Tone* value
- Use State1FunctionCode2 parameter to enter *Space Tone* value
- Define Control Point with RemoteAddress set to 0. Use this control point to operate switch relay.

### Telescada ARS Protocol

Following are the Telescada ARS protocol addressing options for the various *device types* supported by Telescada CCU7C transmitter controller:

#### - Ccu7CCtl

Internal use only, do not use.

#### - TscrTUShortAddr

This is a Telescada RTU. Addressing is short addressing, no routing.

Address: 16 bits

Address2: Not used

RouteNo: Not used

### - TscRTULongAddr

---

This is a Telescada RTU. Addressing is long addressing, no routing.

Address: 24 bits

Address2: Not used

RouteNo: Not used

### - TscRTURelayAddr

---

This is a Telescada RTU. Addressing is short addressing with routing.

Address: 16 bits, RTU address

Address2: 16 bits, first repeater address

RouteNo: 8 bits, route number

### - MotorolaDigital

---

This is a Motorola digital switch.

Address: 24 bits, PC1 = Plug Code 1, PC2 = Plug Code 2, PC3 = Plug Code 3 (hex values)

|      |            |           |       |
|------|------------|-----------|-------|
| Bits | 23 -<br>16 | 15 -<br>8 | 7 - 0 |
|      | PC1        | PC2       | PC3   |

Address2: 16 bits, FC1 = Function Code 1, FC2 = Function Code 2 (hex values)

|      |           |       |
|------|-----------|-------|
| Bits | 15 -<br>8 | 7 - 0 |
|      | FC1       | FC2   |

RouteNo: Not used.

### - GE REMS 101/102

---

This is a GE REMS 101/102 protocol switch.

Address: 12 bits, A = Address Bits

|      |           |           |       |       |
|------|-----------|-----------|-------|-------|
| Bits | 15-<br>12 | 11 -<br>8 | 7 - 4 | 3 - 0 |
|      | 0         | A         | A     | A     |

Address2: 4 bits, FC1 = Function code 1

|      |       |
|------|-------|
| Bits | 3 - 0 |
|      | FC1   |

RouteNo: 4 bits, FC2 = Function code 2

|      |       |
|------|-------|
| Bits | 3 - 0 |
|      | FC2   |

## - SA REMS 101/102

---

This is a Scientific Atlanta REMS 101/102 protocol switch.

Address: 12 bits, A = Address Bits

|      |        |       |       |
|------|--------|-------|-------|
| Bits | 11 - 8 | 7 - 4 | 3 - 0 |
|      | A      | A     | A     |

Address2: 4 bits, FC1 = Function code 1

|      |       |
|------|-------|
| Bits | 3 - 0 |
|      | FC1   |

RouteNo: 4 bits, FC2 = Function code 2

|      |       |
|------|-------|
| Bits | 3 - 0 |
|      | FC2   |

## - SaDigital

---

This is a Scientific Atlanta digital switch.

Address: 16 bits, A = Address 1, PC2 = Address 2

|      |        |       |
|------|--------|-------|
| Bits | 15 - 8 | 7 - 0 |
|      | A1     | A2    |

Address2: Not used.

RouteNo: Not used.

## - FpDigital

---

This is a Fisher Pierce digital switch.

Address: 9 bits, A = Address Bits

|      |      |   |       |       |
|------|------|---|-------|-------|
| Bits | 15-9 | 8 | 7 - 4 | 3 - 0 |
|      | 0    | A | A     | A     |

Address2: 4 bits, T = tone channel

|      |       |       |
|------|-------|-------|
| Bits | 7 - 4 | 3 - 0 |
|      | 0     | T     |

RouteNo: 8 bits, Function codes

|      |       |       |
|------|-------|-------|
| Bits | 7 - 4 | 3 - 0 |
|      | FC2   | FC1   |

## - Sa206

This is a Scientific Atlanta 206 switch.

Address: 32 bits, S = Start Bits, A = Utility Address, B = Switch Address

|      |            |            |            |            |            |           |       |       |
|------|------------|------------|------------|------------|------------|-----------|-------|-------|
| Bits | 31 -<br>28 | 27 -<br>24 | 23 -<br>20 | 19 -<br>16 | 15 -<br>12 | 11 -<br>8 | 7 - 4 | 3 - 0 |
|      | 0          | S          | 0          | A          | 0          | B         | B     | B     |

Address2: 16 bits, A = Function Code 1, B = Function Code 0 (hex values)

|      |            |           |       |       |
|------|------------|-----------|-------|-------|
| Bits | 15 -<br>12 | 11 -<br>8 | 7 - 4 | 3 - 0 |
|      | 0          | A         | 0     | B     |

RouteNo: Not used.

## TS-11 protocol

TS-11 protocol is used by some older load management and capacitor control switches. For compatibility reasons it was also implemented on Telescada NeXGen CCURX capacitor controller.

To configure SmartMaster to communicate with a TS-11 device proceed as follows:

- Define a *Socket* type binding layer client, set IP address to that of computer running the ToneGen program.
- Define a *TS11* type application layer client (no configurable parameters).
- Define an output manager, set *CommChannel* to above application layer client.
- Define remote station of type TS11, and set *OutputChannel* to above output channel.

TS-11 devices have four available function codes (0, 1, 2, 3), each of which performs a device specific function, such as CLOSE, TRIP, etc. Please refer to device user manual. In SmartMaster,

- Define a momentary control output.

- Specify *StationName* parameter so it points to associated TS-11 device.
- Specify *RemoteAddress* parameter to one of four values: 0, 1, 2, or 3.

Executing above momentary control output generates a TS-11 protocol message with function code equal to remote output *RemoteAddress*.

For example, suppose there is a TS-11 device with CLOSE command associated with function code 1. To configure SmartMaster create a momentary control output with *RemoteAddress* equal to 1.

There is no need to specify momentary control output *PulseDuration* parameter as TS-11 messages do not specify pulse duration. The latter is defined in the target instrument itself.

## I/O Points

---

In SM, *I/O points* are subsystems which manage data associated with corresponding data sources and data consumers in the field. Each input has two value registers: *AValue* (actual value), and *CValue* (computed value). The first holds the value found at remote data source / consumer, preserving remote register encoding. The second holds the computed value when one is generated.

There are two kinds of I/O points:

- *Inputs* – manage data associated with corresponding data sources in the field, such as voltage readings, current readings, etc.
- *Outputs* - manage data associated with corresponding data consumers in the field, such as relay coils on switches, analog outputs, etc.

## Definition

---

To define a new I/O point:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Open *I/O Points* window
- Right-click anywhere in the *I/O Points* display
- Execute *New Input...* command to define a new input, or *New Output...* command to define a new output
- Enter new object *name*, do not use two underscore characters (\_\_) in point names, that character sequence is used internally to separate I/O Point names from associated remote station names.
- Enter new object type
- Enter the name of I/O group that the new I/O point will belong to
- Enter new I/O point's Actual Value SQL Type (See below)
- Enter new I/O point's Computed Value SQL Type (See Below)

The new I/O point appears in the *I/O Points* display.

A new point can also be created by copying an existing one. This is useful when creating a number of points that require lots of careful configuration, but differ only in remote station association. Point *RemoteAddress* parameter can stay the same as long as points belong to different remote stations. Point *LocalAddress* parameters, however, need to be distinct, but these are only defined for points reported by a local communications server. To copy an existing I/O point, proceed as follows:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Open I/O points window
- Right-click on I/O point to copy
- Enter new I/O point name

The new I/O point appears in the *I/O Points* display.

## Configuration

---

To configure the new manager first assign I/O point to a remote station, unless I/O point is a stand-alone point not associated with any remote station. Proceed as follows:

- Right-click on the new object
- Execute *Configuration* command
- Modify *StationName* parameter, this associates I/O point with specified remote station
- Select *Save + Exit*, this saves the re-configured I/O point and also changes its name to <Station Name>\_\_<I/O Point Name>, where ' \_\_ ' is the two underscores character sequence mentioned above.

Above modification changes I/O point name. Only after above association has been defined, modify other parameters so all changes can be properly logged using the new object name:

- Right-click on the new object
- Execute *Configuration* command
- Modify parameters as needed. At the very least one must set
  - The value the new object *Manager to Enabled*
  - *RemoteAddress* – parameter defines point address / index on remote station. See *Evaluation* section below
  - *LocalAddress* – parameter defined point local address used by locally defined communications server(s). See below.
  - *VERS* – Value Encoding on Remote Station. (Unsigned Integer, Signed Integer, Twos Complement, or IEEE Floating Point)
  - *WORS* - Width, in bits, On Remote Station
  - *PSP* – Protocol Specific Parameter – see below.
  - *DataRecording* – must be set to *True* if SM is to record this point historical values

- *UseCValue* – Parameter must be set to *True* if I/O point value is computed, and computed value (CValue) is to be used as I/O point value. See *Evaluation* section below
- *Evaluator* – parameter assigns evaluator used to evaluate this point's value. See *Evaluation* section below
- *EvaluationPriority* – parameter determines the order in which points are evaluated.
- Select *Save + Exit*

## PSP Parameter

This parameter is shared by all types of I/O points. It is interpreted differently, depending on communications protocol used to communicate with remote station hosting the I/O point. The table below describes parameter interpretation on a case by case basis.

|                         | Generic DNP 3.0 | Generic Modbus   | Telescada NeXGen Modbus  |
|-------------------------|-----------------|--|--|
| Analog Inputs           | DNP Class.      |  |  |
| Binary Status Inputs    | DNP Class.      | A single bit mask which identifies the status bit. If that bit is 1, I/O point value = 1, else I/O point value = 0.  | A single bit mask which identifies the status bit. If that bit is 1, I/O point value = 1, else I/O point value = 0.  |
| Multi-Val Status Inputs | DNP Class.      |  |  |
| Pulse Counters          | DNP Class.      |  |  |
| Binary Control Outputs  | DNP Class.      | Operational bit patterns, as dictated by target device specification.<br>Bits 0 - 15 are used to define bit pattern to set control to state 0.<br>Bits 16 - 31 define bit pattern to set control to state 1.             | Bits 0 - 15 are used to define bit pattern to set control to state 0.<br>Bits 15 - 8 = 0; Bits 7 - 0 = 0 (See CCDC Technical Notes).<br>Bits 16 - 31 are bits used to define bit pattern to set control to state 1.<br>Bits 31 - 24 = 1; Bits 23 - 16 = 0 (See CCDC Technical Notes).<br>PSP = 16777216, or 256 (Reverse operation logic). |
| Timed Control Outputs   | DNP Class.      | Operational bit patterns, as dictated by target device specification.<br>Bits 0 - 15 are used to define bit pattern to set control to state 0.<br>Bits 16 - 31 are used to define bit pattern to set control to state 1. | Bits 0 - 15 are used to define bit pattern to set control to state 0.<br>Bits 15 - 8 = 0; Bits 7 - 0 = 1 or 2 (See CCDC Technical Notes).<br>Bits 16 - 31 are bits used to define bit pattern to set control to state 1.<br>Bits 31 - 24 = 1; Bits 23 - 16 = 1 or 2 (See CCDC Technical Notes).  |
| Mom. Control Outputs    | DNP Class.      | Operational bit patterns, as dictated by target device specification.<br>Bits 0 - 15 are to define bit pattern to operate control.   | Bits 15 - 8 = Pulse duration, ms; Bits 7 - 0 = 0x03 or 0x04 (See CCDC Technical Notes).  |

### Notes:

- Protocols not listed below do not use the parameter, i.e. parameter is ignored.
- In all cases, bit 0 is the least significant bit.
- CCDC Technical Notes is the NeXGen firmware technical manual available from Telescada.
- Empty table cell means parameter is ignored in that case.



---

## Evaluation

---

Each I/O point manager holds two values:

- Actual value, stored in point's *AValue* register – this is unscaled value as reported by remote station, and
- Computed value, stored in point's *CValue* register, this is a value computed from point's *AValue*, and / or other values in the database.

Evaluators are short data conversion programs, written by SM administrators in T-SQL, which compute *CValue* from *AValue* and, possibly, from other values. Please see the section on Evaluators below.

Every poll cycle is followed by an evaluation cycle where point computed values are updated. The order in which this happens is dependent on values of I/O point *EvaluationPriority* parameters. For every I/O point, the lower the *EvaluationPriority* parameter value, the higher the point's evaluation priority. Evaluation order is important when results are interdependent. For example, when calculating power factor, kW and VAr numbers should be available before power factor calculation takes place.

If parameter *UseCValue* is set to True, I/O point's *CValue* is used as I/O point's value. In this case the following I/O point parameters need to be defined as follows:

- *UseCValue* = *True*
- *RemoteAddress* – see below.
- *Evaluator* – parameter must be defined, and not equal to *NONE*.

In the opposite case, if parameter *UseCValue* = False, I/O point's *AValue* is used as I/O point's value. In this case the following I/O point parameters need to be defined as follows:

- *UseCValue* = *False*
- *RemoteAddress* >= 0
- *Evaluator* = *NONE*.

If remote station poll is to include a query for I/O point value, that I/O point value *RemoteAddress* parameter value must be >= 0. This populates I/O point's *AValue* register. One can then use *AValue* as I/O point value, or use an evaluator which uses *AValue*, and possible other values, to compute point's *CValue*.

On the other hand, if I/O point is a purely computed one (only *CValue* is of interest), then *RemoteAddress* parameter value must be -1. In this case remote station poll does not include a query for I/O point value.

---

## Evaluators

---

Evaluators are short data conversion programs, written by SM administrators in T-SQL. When evaluator is mapped to an I/O point via point's Evaluator parameter, the point's CValue is computed from point's AValue, and possibly from other values, as dictated by the said Evaluator.

To define a new Evaluator:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Open *Evaluators* window
- Right-click anywhere in the *Evaluators* display
- Execute *New Evaluator...* command
- Enter new object's *name*

The new Evaluator appears in the *Evaluators* display.

To configure the new manager:

- Right-click on the new object
- Execute *Configuration* command
- Modify parameters as needed. At the very least one must set
  - The value the new object *Manager* to *Enabled*
  - *SourceCode* – must be written in T-SQL language (see below)
- Select *Save + Exit*

Alternatively, evaluator already associated with an I/O point can also be edited as follows:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Open *I/O Points* window
- Right-click on *I/O point of interest*
- Execute *Configuration – I/O Point* command
- Press and hold down the *keyboard Ctrl* key
- Click on *Evaluator* parameter

Above procedure opens configuration editor and loads parameters of evaluator associated with I/O point. Evaluator parameters can now be edited as described above. **Caution:** If evaluator so edited is associated with multiple I/O points, all those points are affected when evaluator is changed.

---

## - Source Code Authoring

---

SM Administrators write Evaluator source code in T-SQL language. For convenience, SM generated source code provides for a number of variables which can be used by Evaluator author without declaration, including:

- @Name (varchar(64)) – Name of input executing the Evaluator
- @AValue (float) – Input current actual value
- @OldAValue (float) – Input previous value of @AValue
- @CValue (float) – Input current value of @CValue (about to be computed)
- @OldCValue (float) – Input previous value of @CValue
- @Type (varchar(64)) – I/O point type (analog, status, etc.)
- @SQLType (varchar(64)) – I/O point SQL type (int, bit, float, etc)
- @Input (bit) – True means I/O point is an input, False means it is an output
- @DataRecording (bit) – True means I/O point data recording is enabled
- @UseCValue (bit) – this variable is either true or false. When true, input CValue is used as point's value, when false input's AValue is used as point's value.
- @AValueDT (datetime2) – Date and time of @AValue
- @OldAValueDT (datetime2) – Date and time of @OldAValue
- @OldCValueDT (datetime2) – Date and time of @OldCValue
- @SourcePath (varchar(128)) – I/O point source path
- @StationName (varchar(64)) – Name of remote station associated with I/O point
- @RemoteAddress – I/O point parameter RemoteAddress value
- @CanCtl (bit) – See section *Dynamic Operation Control* below.

In the simplest case the author can use @CValue to store result of his / her computation. This computation can involve variables listed above. For convenience, SM simply pre-fetches those values before calculation, and saves @CValue after calculation. In more elaborate calculations, if other variables need to be used, they have to be read from database explicitly. Also, if other variables need to be saved, they need to be written into database explicitly.

For example, to simply rescale I/O point's actual value by a factor of 10, use the following:

```
SET @CValue = @AValue * 10
```

As another example, to calculate the difference between the new and old values, proceed as follows:

```
SET @CValue = @AValue - @OldAValue
```

As a third example, suppose there are a number of remote stations each of which is reporting power on phases A, B, C via I/O points <...>\_\_KWA, <...>\_\_KWB, and <...>\_\_KWB, respectively. Here, <...> stands for remote station name hosting above analog inputs. Suppose one needs to create a purely computed I/O point on each remote station which displays total power consumed on all three phases. One could write one Evaluator for each of the remote stations, in each case read in power from the three phases, and sum them up. Instead, one can use only one evaluator for all substations (lines which start with two dashes are comments ignored by compiler):

```
-- Declare local variables
DECLARE @APW float = 0
DECLARE @BPW float = 0
DECLARE @CPW float = 0
DECLARE @InputName varchar(128)

-- Read in power from phase A
SET @InputName = @StationName + '__KWA'
SELECT @APW = AValue
FROM [RSExec].[IOPoints]
WHERE Name = @InputName

-- Read in power from phase B
SET @InputName = @StationName + '__KWB'
SELECT @BPW = AValue
FROM [RSExec].[IOPoints]
WHERE Name = @InputName

-- Read in power from phase C
SET @InputName = @StationName + '__KWC'
SELECT @CPW = AValue
FROM [RSExec].[IOPoints]
WHERE Name = @InputName

-- Compute Total Power
SET @CValue = @APW + @BPW + @CPW
```

The author may opt to use any other variables from SM database; however those need to be read into memory explicitly by authored code.

Note also that source code can include calls to stored procedures associated with SMPrograms (see SCADA section).

For a list of tables of interest when writing code for SM please consult *Reports Tables* technical note.

---

## - Source Code Compilation

---

Once the *SourceCode* parameter has been defined (i.e. Evaluator source code has been written), save the modified Evaluator by pressing the *Save + Exit* button. SM saves Evaluator in database. When Evaluator is enabled SM tries to compile the source code. If Evaluator compiles successfully the *Compiled* column in *Evaluators* display shows a *Yes* value, otherwise the value displayed is *No*. In addition, an appropriate event log entry is generated.

---

## OVM (Output Value Management)

---

OVM is driven by administrator written program(s) in SCADA service (see SCADA section) and / or evaluators in RemoteStations service. A program computes I/O point's CValue, and *RemoteStations* service *OutputManager* subsystem whose OVM parameter is enabled, sends appropriate control code to output point in the field, so associated I/O point CValue and AValue are in congruence.

For an output point to be involved in OVM, its *OVM* parameter must be *Enabled*, and its *StationName* parameter must point to a valid remote station identifier.

---

## Dynamic Operation Control

---

Output availability for control can be manipulated dynamically by changing the value of *@CanCtl* I/O point parameter in Evaluators. When *@CanCtl* value is 0 (False) output is not available for operation, manually or automatically by any of SM algorithms. Parameter value is ignored by inputs.

---

## I/O Point Groups

---

I/O points can be bundled into groups. Users can filter I/O points in *I/O Points window* by groups. All I/O points in a group can also be enabled or disabled by enabling or disabling their group.

To define a new I/O point group:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click anywhere in the *I/O Point Groups* display
- Execute *New I/O Group...* command
- Enter new group's *name*

The new I/O group appears in the *I/O Point Groups* display.

To reconfigure a group:

- Right-click on the group of interest
- Execute *Configuration* command
- Modify parameters as needed.
- Select *Save + Exit*

To make an I/O point a member of a group, proceed as follows:

- Right-click on I/O point of interest
- Execute *Configuration* command
- Set *I/OGroup* parameter by selecting appropriate I/O group from the list of defined groups
- Select *Save + Exit*

## Remote Station Groups

---

Remote stations can be bundled into groups. Users can filter remote stations in *Remote Stations service window* by groups. All remote stations in a group can also be enabled or disabled by enabling or disabling their group.

To define a new remote stations group:

- Start the *Main Viewer*
- Open *RemoteStations* service window
- Right-click anywhere in the *Remote Station Groups* display
- Execute *New Remote Station Group...* command
- Enter new group's *name*

The new remote stations group appears in the *Remote Stations Groups* display.

To reconfigure a group:

- Right-click on the group of interest
- Execute *Configuration* command
- Modify parameters as needed.
- Select *Save + Exit*

To make a remote station a member of a group, proceed as follows:

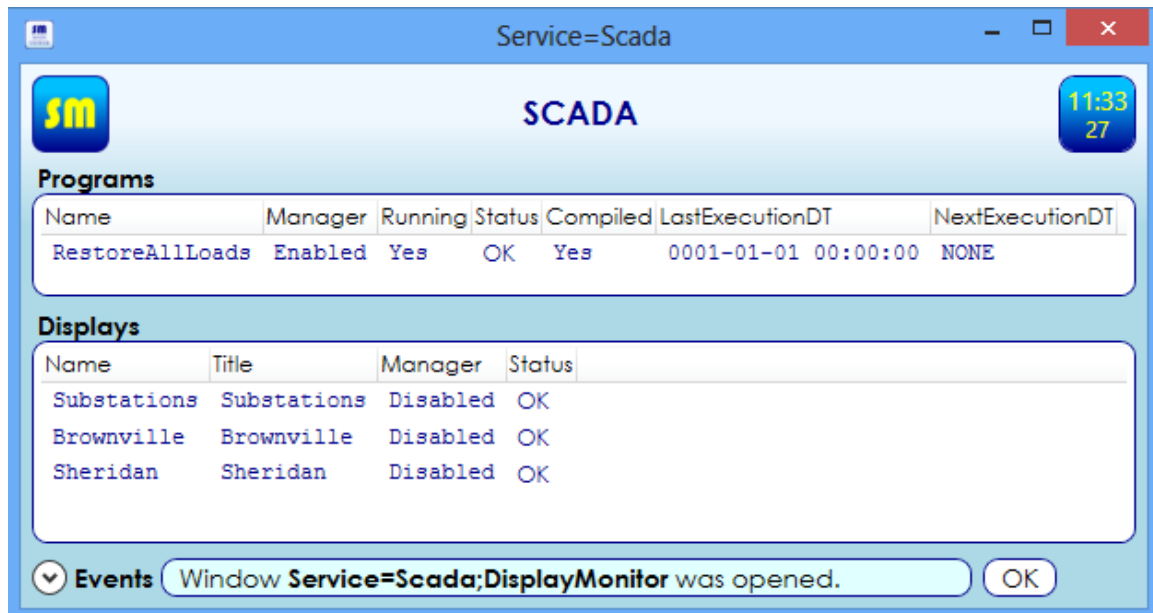
- Right-click on remote station of interest
- Execute *Configuration* command
- Set *RSGroup* parameter by selecting appropriate remote station group from the list of defined groups
- Select *Save + Exit*

## SCADA

SCADA service is an application which depends on RemoteStations and Communications services, as well as those of DBMS. Before SCADA functions can be configured RemoteStations and Communications must be fully configured.

In SM, SCADA service provides two functions:

- Programs - Written by SM administrators
- One-Line Diagram Displays – Interactive one-line diagrams of the system managed by SM



## Programs

SCADA programs are written in T-SQL. As opposed to Evaluators, SCADA programs are completely open, unrestricted, and provide complete freedom as to which variables are used as program inputs and outputs. With that freedom come dangers and responsibilities that program authors must be aware of. While one can use any database variables as inputs, it is strongly advised that **only I/O point CValue registers are used for program output.**

When using programs to do other tasks – such as, for example, global database value changes, it is advisable to disable all services affected by such programs. Failure to disable services increases the risk of database concurrencies.

Programs are meant to execute control algorithms and logic authored by SM administrators. It is not advisable to use programs for simple I/O point scaling & calculations – Evaluators already do that.

---

## Program Creation

---

To create a new program, proceed as follows:

- Open SCADA service window
- Right-click onto *Programs* display
- Execute *New Program...* command
- Enter new program *name*
- Select *Save + Exit*

The new program appears in the *Programs* display. To configure the new program:

- Right-click on the new program
- Execute *Configuration – Program* command
- Modify parameters as needed. At the very least one must set the value the new object *Manager* to *Enabled* and write program source code (*SourceCode* parameter).
- Select *Save + Exit*

If the program compiles the *Compiled* column in *Programs* display is set to *Yes*, else it is set to *No* and alarm is generated.

One should always make sure that the program compiles and works as intended before it is set for automatic, scheduled execution. This should be done by first manually executing the program as described in the User Manual, and observing program execution results.

---

## Database Tables

---

Program input comes from parameters in system tables. The DBMS contains many tables and other system objects. To locate SM related tables and parameters contained within proceed as follows:

- Start *Microsoft SQL Server Management Studio (SSMS)*
- Log on using your *SM account*
- Open *Object Explorer*
- Expand *Databases*
- Under *Databases* expand the database bearing the same name as your *system site code*
- Under the database expand *Tables*

Above lists the tables used by SM. It is strongly advised that no modifications to tables or table column values be done via SSMS.



---

## Table Columns

---

Table columns are SM configuration parameters. Parameter names can be referenced by their names in SM programs. Parameter values are thus used as input for SCADA service programs. When referencing parameter name in SM programs, it is advised that their full names

Schema.Table.Column

are used, where *Schema* is DBMS security schema associated with the table, *Table* is the name of the table of interest, and *Column* is the name of the column of interest.

For example, the full name of I/O Point CValue register is

RSExec.IOPoints.CValue

Above, *RSExec* is the security schema name, *IOPoints* is the table name, and *CValue* is the column name.

To get a list of columns in a specific table proceed as follows:

- Start *Microsoft SQL Server Management Studio (SSMS)*
- Log on using your *SM account*
- Open *Object Explorer*
- Expand *Databases*
- Under *Databases* expand the database bearing the same name as your *system site code*
- Under the database expand *Tables*
- Under *Tables* expand the *table of interest*
- Under the table expand *Columns*

---

## Program Execution

---

Programs can be executed manually as described in the User Manual. They can also be configured to run automatically by SCADA service on a periodic basis. Proceed as follows:

- Right-click on the program of interest
- Execute *Configuration – Program* command
- Modify parameters *ExecutionPeriod* and *OffsetFromMidnight* parameters as needed.
- Select *Save + Exit*

Parameter *ExecutionPeriod* specifies how often the program should execute (in seconds); the value of zero disables periodic execution.

Parameter *OffsetFromMidnight* specifies the number of seconds above time grid should be offset from last midnight.

Periodic program execution can also be disabled by setting the *Manager* parameter to *Disabled*.

Programs can also be cascaded, i.e. one program can call (execute) other programs by executing their associated stored procedures. Each program is stored in the DBMS as a stored procedure named as follows:

SMPrograms.SP\_<name>

Above, <name> is the *name of the program* as listed in the SCADA service *Programs* display. Program stored procedures have no arguments.

Programs can also call Evaluators executing their associated stored procedures. Each Evaluator is stored in the DBMS as a stored procedure named as follows:

Evaluators.SP\_<name>

Above, <name> is the *name of the Evaluator* as listed in the RemoteStations service *Evaluators* display. Evaluators are specialized programs associated with I/O points mapped to them. Consequently, evaluator stored procedures have one argument, I/O point number:

RSExec.IOPoints.Number

which is of (SQL) *bigint* type (signed 64 bit integer). This number uniquely identifies the I/O point.

Beware of circular references. A circular reference, in its simplest form, is when program A calls program B, and program B calls program A. When a program gets into circular reference an endless loop occurs, results are unpredictable.

## One-Line Diagram Displays

---

User Manual describes how to open and view one-line diagram displays (OLDDs). This section describes how to create OLDDs.

Each OLDD is a collection of display elements (DEs) created by SM administrator. Some of those DEs are usually mapped onto respective I/O points so they display real time information and provide interactivity with SM. For example, a DE can be mapped onto a control output I/O point. That DE can show the state of that output in real time, as well as provide commands to operate the output right from the OLDD.

DEs can be atomic or composite. In the latter case they are collections of simpler DEs. To create an OLDD, the top DE must be declared as a display by setting DE parameter *IsDisplay* to *Yes*.

## Display Designer

---

OLDDs are created via Display Designer (DD). To open DD proceed as follows:

- In *Main Viewer* open *SCADA service* window
- Right-click on *SCADA service* window title and execute *Display Designer* command

DD has two displays, Canvas and Library. Canvas is where display elements (DEs) are designed, and Library is where they are stored for later use in building more complex DEs. This library is a local (customer server resident) library. There exists another library on BTE server, called central library, which contains DEs designed by BTE. Those DEs can be downloaded into local library as described later in this section. Local library holds locally designed DEs and DEs downloaded from central library.

## Display Element Types

---

There are two types of DEs: atomic and composite.

### - Atomic Display Elements

---

These are DEs which contain no children, i.e. they are not made of any other DEs. Current list of atomic DEs contains the following DE types:

|          |   |
|----------|---|
| Line     | Straight line, two end points   |
| Polyline | Collection of interconnected straight line segments, no limit on number of segments. Polyline start and end points are distinct (open figure) |
| Polygon  | Collection of interconnected straight line segments, no limit on number of segments. Polyline start and end points match (closed figure)      |
| Circle   | Circle of arbitrary radius  |
| Arc      | Arbitrary open curve  |
| Gauge    | DE which changes appearance based on associated I/O point   |
| Text     | DE for displaying text and gauges displaying text   |

### - Composite Display Elements

---

These are DEs made of other DEs (children). Multiple generations of children are allowed. This allows for building of more complex DEs and, ultimately, OLDDs.

## Display Element Creation

---

To create a new DE proceed as follows:

- Open Display Designer
- Right-click anywhere in the Canvas
- Select *New Display Element* command
- Select DE type from the list

After this procedure depends on the type of DE one is building.

### - Line

---

- Click in the *Canvas* where first point is supposed to be located
- Click in the *Canvas* where second point is supposed to be located
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

### - Polyline, Polygon

---

- Click in the *Canvas* where first point is supposed to be located
- Repeat above until all points have been defined
- Right-click anywhere in the *Canvas*
- Execute *Done Adding Points* command
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

### - Circle

---

- Click in the *Canvas* where first point on circle perimeter is supposed to be located
- Repeat above three more times until the circle appears
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command

- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

### - Arc

---

- Click in the *Canvas* at three different locations
- Drag the points to their intended final locations, this manipulates arc shape
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

### - Text

---

- Click in the *Canvas* at where the top left text corner is supposed to be located
- Click in the *Canvas* at where the bottom right text corner is supposed to be located
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

### - Date / Time Display

---

- Create a text element
- Configure text element as follows:
  - Set ShowText parameter to True
  - Set Text parameter to #DATE# to display current date
  - Set Text parameter to #TIME# to display current time
  - Set Text parameter to #DATE#TIME# to display current date and time

This saves the new DE to local library.

---

### - Gauge – Relay\_02 (Two-state relay)

---

- Click in the *Canvas* at four different places in approximately straight line. The middle line segment opens / closes when DE is mapped to a control output, matching that output state
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

---

### - Gauge – Relay\_03 (Three-state relay)

---

- Click in the *Canvas* at four different places in approximately straight line. The middle line segment opens left, opens right, or closes when DE is mapped to a tristate status input or control output, matching that input / output state
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

---

### - Gauge – VBREAKER\_02 (Two-state breaker, vertical orientation)

---

- Click in the *Canvas* at four different places in approximately straight line. The middle line segment opens / closes when DE is mapped to a control output, matching that output state
- Drag the points to their intended final locations
- Right-click anywhere in the *Canvas*
- Execute *Attach to Grid* command
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

---

## - Composite

---

- Right-Click on a DE in the library
- Execute *Add Display Element to Canvas* command – DE appears in *Canvas*
- Drag DE to desired location in *Canvas*
- Size DE to desired size (see below for procedure)
- Repeat above steps to add as many DEs to *Canvas* as desired
- Right-click anywhere in the *Canvas*
- Execute *Save to Library...* command
- Enter the new DE *name*

This saves the new DE to local library.

---

## Change Display Element

---

User authored DEs can be changed in two ways: editing and reconfiguration.

---

## - Edit Display Element

---

Editing DE means to change DE appearance. To edit DE proceed as follows:

- Open *Display Designer*
- Right-Click on the DE of interest in the Library
- Execute *Edit Display Element...* command

This places the DE onto the Canvas. If DE is atomic one can drag DE points to new locations. If DE is composite one can drag DE children to new locations. Either way DE appearance changes as a consequence. When done the modified DE can be saved as follows:

- Right-Click anywhere in Canvas
- Execute *Save Modifications...* command
- Confirm command execution

Alternatively one can save DE as a Display by executing the *Save Modifications as Display...* command. This saves DE modifications in library and makes DE an OLDD.

DEs downloaded from central library cannot be edited. They can be edited once used as children of a composite DE.

---

## - Configure Display Element

---

Configuring DE means changing DE parameters. To configure DE proceed as follows:

- Open *Display Designer*
- Right-Click on the DE of interest in the Library
- Execute *Configuration - Display Element...* command
- Change parameters of interest
- Select *Save + Exit*

DEs downloaded from central library cannot be reconfigured. They can be configured once used as children of a composite DE.

---

## Change Display Element Children

---

To change (edit or reconfigure) a DE child proceed as follows:

- Open *Display Designer*
- Right-Click on the DE of interest in the Library
- Execute *Show Children* command

At this point DE children are visible in the library. They can be changed like any other DE as described in previous sections.

---

## Display Element Colors

---

Each DE is assigned default colors. Colors can be changed by reconfiguring the DE. Each DE has six color parameters: Two stroke colors (Parameters SC0 and SC1), and four fill colors (FC0 – FC3).

---

### - Stroke Colors

---

Stroke colors are used to color line elements and text letters. All DEs are colored using stroke color SC0 when DE is *unmapped* or is mapped to an object in *normal* state, and SC1 when DE is mapped to an object in *alarm* state.

---

### - Fill Colors

---

Fill colors are used to color DEs which form two dimensional areas bordered by stroke colored line elements.

All DEs are colored using fill color FC0 when unmapped, or when mapped to I/O point of unknown type or state.



All DEs are colored using standard alarm brush (alternating red / yellow) when mapped to objects in alarm state.

When mapped to binary *status* or any of the *control* outputs, fill colors FC1 and FC2 are used for the two possible I/O point states.

When mapped to a multi-valued *status* point, Fill color FC1 is used when status point is in state 0, and FC2 is used when in any of the other possible states.

When mapped to an *analog* point, fill colors are used as follows:

|         |  |
|---------|--|
| F<br>C1 | Analog point in <i>low</i> or <i>very low</i> region   |
| F<br>C2 | Analog point in <i>normal</i> region                   |
| F<br>C3 | Analog point in <i>high</i> or <i>very high</i> region |

## Mapping of Display Elements

Display elements can be mapped onto I/O points via DE parameter *ObjectName*. Associated I/O point can monitor any of database items of the following possible types:

| <u>Object Type</u> | <u>Description</u>   |
|--------------------|--|
| DemandResponder    | Load management demand responder                             |
| IOPoint            | RemoteStations service I/O point                             |
| IOPoint2           | RemoteStations service I/O point observing another I/O point |
| LoadGroup          | Load management load group                                   |
| LoadManagement     | Load management subsystem                                    |
| LSRManager         | Load state reinserting manager                               |
| OutputManager      | RemoteStations output manager                                |
| SystemUsers        | System users subsystem                                       |

## Text Display Element Gauging

Text display element can be displayed not only as a rectangle, but as a gauge. This functionality is controlled by setting parameter *Gauging*. At this time only one option is available: vertical bar, the length of which depends on the value of associated mapped I/O point.

## Load Management

LoadManagement service depends on RemoteStations and Communications services, as well as those of DBMS. Before LoadManagement functions can be configured RemoteStations and Communications must first be fully configured.

In SM, LoadManagement service provides the following functions:

- Peak shaving
- Load state reinserting
- Load rotation

**Service**

| Name           | Manager | Running | Status | Strategy | LMWaiver | LMWStartTOD | LMWEndTOD |
|----------------|---------|---------|--------|----------|----------|-------------|-----------|
| LoadManagement | Enabled | Yes     | OK     | Primary  | Enabled  | 11:00:00    | 22:00:00  |

**Programs**

| Name         | Manager | Running | Status |
|--------------|---------|---------|--------|
| PeakShaver   | Enabled | Yes     | OK     |
| AfterPeakLSR | Enabled | Yes     | OK     |
| EveningLSR   | Enabled | Yes     | OK     |

**Load Groups**

| Name   | Manager | Status | MngStartTOD | MngEndTOD |
|--------|---------|--------|-------------|-----------|
| MoWeFr | Enabled | OK     | 10:00:00    | 22:00:00  |
| TuThSa | Enabled | OK     | 10:00:00    | 22:00:00  |
| Sunday | Enabled | OK     | 10:00:00    | 12:00:00  |

**Demand Inputs**

| LMDIName | Manager | LMDIStatus | Status | ValueDT             | Value | Comments |
|----------|---------|------------|--------|---------------------|-------|----------|
| Blair    | Enabled | OK         | OK     | 2013-12-27 12:24:00 | 349   | OK       |
| Pender   | Enabled | OK         | OK     | 2013-12-27 12:24:00 | 332   | OK       |
| Pierce   | Enabled | OK         | OK     | 2013-12-27 12:24:00 | 350   | OK       |

**Load Points**

| LMLPName   | Manager | LMLPStatus | AOT     | Status | ValueDT             | Value | Comments    |
|------------|---------|------------|---------|--------|---------------------|-------|-------------|
| AlbersJ    | Enabled | OK         | 0:00:00 | OK     | 2013-12-27 12:24:00 | OFF   |             |
| ReppertS   | Enabled | OK         | 0:00:00 | OK     | 0001-01-01 00:00:00 | ON    |             |
| StratmanE  | Enabled | OK         | 0:00:00 | OK     | 0001-01-01 00:00:00 | ON    | MANUAL MODE |
| TimmermanF | Enabled | OK         | 0:00:00 | OK     | 0001-01-01 00:00:00 | ON    |             |

**Events** Connected to \\POINCARE\SmartMaster service, site NYBTE.

The service uses *demand inputs* to account for managed system power demand, and *load points* to manage demand response. Each load point can be used to connect / disconnect one or more power consumers. Load point operations can be manually initiated by an operator, or automatically executed by SM based on *program* settings. One can create as many programs as necessary. Three types of program managers are available: *peak shavers*, *load state reinserting*

managers, and *load rotation* managers. Finally, load points can be grouped in *load groups* for group management.

To configure *LoadManagement* service proceed as follows:

- Start the *Main Viewer*
- Right-click on *LoadManagement*
- Execute *Configuration – Service* command
- Modify parameters as needed. At the very least one must set
  - *Manager* to *Enabled*.
- Select *Save + Exit*

Alternatively, with *LoadManagement* service window already open, proceed as follows:

- Right-click on window title
- Execute *Configuration – LoadManagement* command
- Modify parameters as needed.
- Select *Save + Exit*

---

## Strategies

The service allows for three possible strategies: *Primary*, *Secondary*, and *Emergency*. One can define a set of load groups for each strategy, and assign each load point to appropriate load group for each strategy as described in *Load Points* section. Switching between strategies can only be manually initiated by an operator.

Single strategy load management is possible – one can leave *Strategy* parameter set to *Primary* and never change it. In this case load point parameters *Group2* and *GroupE* can remain undefined.

---

## Demand Inputs

Demand inputs are used to account for managed system power demand. To create a new demand input proceed as follows:

- Start the *Main Viewer*
- Open *LoadManagement* service window
- Right-click anywhere in the *Demand Inputs* display
- Execute the *New Demand Input...* command
- Enter the new demand input *name*

The new demand input appears in *Demand Inputs* display. To configure the new demand input:

- Right-click on the new demand input

- Execute *Configuration – Demand Input* command
- Modify parameters as needed. At the very least one must set
  - *Manager* to *Enabled*.
  - *AnalogInput* to appropriate *RemoteStations* analog input reading demand at the point of interest
- Select *Save + Exit*

It is advisable to configure associated *RemoteStations demand input* configured so its value is never more than input's *very high* threshold, and never less than input's *very low* threshold. This way out of range values are still reported via alarms when demand input value is either *high* or *low*, but are never so far out of range as to cause inappropriate peak shaver demand response. This behavior can be accomplished by computing demand input value using appropriate evaluator.

---

## Load Points

---

Load points are objects used by *LoadManagement* service to connect and disconnect power users from the power grid. To create a new demand input, proceed as follows:

- Start the *Main Viewer*
- Open *LoadManagement* service window
- Right-click on load management window title and select *Load Points*
- Right-click anywhere in the *Load Points* display
- Execute the *New Load Point...* command
- Enter the new load point *name*

The new load point appears in *Load Points* display. To configure the new load point:

- Right-click on the new load point
- Execute *Configuration – Load Point* command
- Modify parameters as needed. At the very least one must set
  - *Manager* to *Enabled*.
  - *ControlOutput* to appropriate *RemoteStations* control output of interest
- Select *Save + Exit*

Above minimum configuration is enough so the new load point can be used to manually operate associated control output. For automatic operations a few more load point parameters need to be defined:

- *PowerRating* – parameter needs to be defined for all load points that are managed by a peak shaving program.
- *LoadType* – must be set to *Primary* for load point to be managed by any of the programs. See the section on *Secondary Loads* below.
- *Group1* – parameter must be defined for load points involved in *Primary strategy* load management.

- *Group2* – parameter must be defined for load points involved in *Secondary strategy* load management.
- *GroupE* – parameter must be defined for load points involved in *Emergency strategy* load management.
- *DemandResponder* – every load that is to be managed by a demand responder type program must specify that program.
- *LSR* - every load point whose state is to be automatically reinserted by any *LSR* type program must have this parameter set to *Enabled*.

To configure associated *RemoteStations* service control point click on load point *ControlOutputConfiguration* parameter. Control point *ManualMode* parameter must be set to *False*, otherwise LoadManagement service programs ignore (do not operate) that load point.

## Load Groups

---

Load groups are objects for grouping load points. They contain parameters shared by associated load points. A group can also be used to operate all loads assigned to that group. To create a new load group, proceed as follows:

- Start the *Main Viewer*
- Open *LoadManagement* service window
- Right-click anywhere in the *Load Groups* display
- Execute the *New Load Group...* command
- Enter the new load group *name*

The new load group appears in *Load Groups* display. To configure the new load group:

- Right-click on the new load group
- Execute *Configuration – Load Group* command
- Modify parameters as needed. At the very least one must set
  - *Manager* to *Enabled*
- Select *Save + Exit*

Above minimum configuration is enough to run group default settings. Usually one will want to set up days of week when load management is allowed, *AOTBaseDT*, and *AOTMode*. The latter parameter can have one of the following values:

- *CurrentDay* – Current day of week Accumulated Off Time (AOT) only is used
- *EnabledDays* – AOT sum of all management enabled days of week is used

## Manual Load Point Operation

---

To operate load point manually proceed as follows:

- Right-Click on the load point of interest in the *Load Points* window

- Execute *Operate Load Point...* command
- *Confirm* command execution

To operate a number of load points simultaneously:

- Select load points of interest in the *Load Points* window
- Right-Click on any of the selected load points
- Execute *Operate Load Point...* command
- *Confirm* command execution

To operate all load points assigned to the same load group:

- Right-Click on the load group of interest in the *LoadManagement* service window
- Execute *Operate Load Points...* command
- *Confirm* command execution

---

## Load Point Operation Priority

---

Load point 'disconnect' priority is computed in the following order:

- Associated load group DisconnectPriority parameter values are compared. Load points with lower parameter value have higher 'disconnect' priority.
- Load point DisconnectPriority parameter values are compared. Load points with lower parameter value have higher 'disconnect' priority.
- Load point AOT values are compared. Load points with lower AOT value have higher 'disconnect' priority.

Load point reconnect priority is computed in the following order:

- Associated load group ReconnectPriority parameter values are compared. Load points with lower parameter value have higher reconnect priority.
- Load point ReconnectPriority parameter values are compared. Load points with lower parameter value have higher reconnect priority.
- Load point AOT values are compared. Load points with higher AOT value have higher reconnect priority.

---

## Secondary Loads

---

Secondary loads are load points which operate only when associated primary load operates. Secondary loads are not operated directly by any LoadManagement program. A load point is a primary load point when parameter *LoadType* is set to *Primary*. A load point is secondary when parameter *LoadType* is set to identify associated primary load. Beware of circular references. For example, when load A is secondary to load B and vice versa, neither load point will operate automatically under any LoadManagement program.

---

## Programs

---

LoadManagement service offers three types of programs: demand responder, load state reinserting (LSR) manager, and load rotation (LR) manager.

### Demand Response

---

Demand response is done by *DemandResponder* type program(s). One can create and run as many programs as necessary. *DemandResponder* type programs run a peak shaving algorithm which attempts to keep power demand below operator specified high threshold.

To create a new program, proceed as follows:

- Start the *Main Viewer*
- Open *LoadManagement* service window
- Right-click anywhere in the *Programs* display
- Execute the *New Program...* command
- Enter the new program *name*
- Select *DemandResponder* for program type

The new program appears in the *Programs* display. To configure the new program:

- Right-click on the new program
- Execute *Configuration – Program* command
- Modify parameters as needed. For the program to run correctly all parameteres need to be defined properly.
- Select *Save + Exit*

Parameter *TDAI* (Total Demand Analog Input) must be set to identify *RemoteStations* service analog input responsible for reporting total demand managed by the program. *TDAI* threshold parameter values must satisfy the following condition for the program to be able to function:

$$VlowThr \leq LowThr < HighThr \leq VhighThr$$

*TDAI* Low threshold (*LowThr*) and high threshold (*HighThr*) parameter values are used for load management. When *TDAI* demand reading is higher than *HighThr* setting enough load points are disconnected to bring demand below *HighThr* setting if possible. When *TDAI* demand reading is lower than *LowThr* setting enough load points are reconnected to bring demand just above *LowThr* setting if possible. *TDAI* *VlowAlarm*, *LowAlarm*, *NormalAlarm*, *HighAlarm* and *VHighAlarm* parameteres should be set to *False* to avoid unnecessary alarms.

*AnalogInputConfiguration* parameter can be used to configure *TDAI*. By clicking on *AnalogInputConfiguration* one goes straight to *TDAI* configuration window. Alternatively, clicking on *AnalogInput* parameter while holding down the keyboard *Ctrl* key also opens *TDAI* configuration window.

*InputManagerName* must identify RemoteStations service input manager responsible for polling analog inputs which serve as LoadManagement service demand inputs. This way the program executes automatically right after completion if every poll initiated by the input manager. If *InputManagerName* is set to *NONE*, the program may be executed on a fixed schedule, or by operator issued command. To execute a program by command:

- Right-click on the program in the *Programs* display
- Select *Execute management Cycle...* command

It is not advisable to execute *DemandResponder* type program by command more than once in a row. When program execution completes demand readings are still the same as they were before program execution. The readings change only after a poll. Repeated program execution may disconnect too many customers from the power grid.

*ExecutionPeriod* and *OffsetFromMidnight* parameters can both be set. When not equal to 0, they define periodic execution schedule of DemandResponder object. In this case *InputManagerName* parameter should be set to *NONE*.

## Load State Reinserting

---

Load state reinserting (LSR) is done by *LSRManager* type program(s). One can create and run as many programs as necessary. *LSRManager* type programs run an algorithm which attempts to reinsert load states by dispatching appropriate control messages. If a load point is thought to be ON at the time of LSRManager execution, an ON command is sent to that load. If a load point is thought to be OFF at the time of LSRManager execution, an OFF command is sent to that load.

To create a new program, proceed as follows:

- Start the *Main Viewer*
- Open *LoadManagement* service window
- Right-click anywhere in the *Programs* display
- Execute the *New Program...* command
- Enter the new program *name*
- Select *LSRManager* for program type

The new program appears in the *Programs* display. To configure the new program:

- Right-click on the new program
- Execute *Configuration – Program* command
- Modify parameters as needed. For the program to run correctly all parameteres need to be defined properly.
- Select *Save + Exit*

*ExecutionPeriod* and *OffsetFromMidnight* parameters can both be set. When not equal to 0, they define periodic execution schedule of LSRManager object. When set to 0, one of the methods below can be used to schedule object execution.



Program parameter *SchedulingMethod* must be set to *MngStartTOD* if the program is to execute daily at the time of day specified by *MngStartTOD* parameter.

Program parameter *SchedulingMethod* must be set to *MinAfterEndOfPP* if the program is to execute daily *MinAfterEndOfPP* minutes after peak period.

Program parameter *OperationMode* must be defined appropriately to specify which load points are to be issued LSR commands. The table below lists available options:

|                       |   |
|-----------------------|---|
| OffAndOn              | All load points are issued LSR commands   |
| OffOnly               | Only loads in OFF state are issued LSR commands   |
| OnOnly                | Only loads in ON state are issued LSR commands  |
| OffAndOnOperatedToday | Only loads operated previously on the day of program execution are issued LSR commands              |
| OffOnlyOperatedToday  | Only loads in OFF state operated previously on the day of program execution are issued LSR commands |
| OnOnlyOperatedToday   | Only loads in ON state operated previously on the day of program execution are issued LSR commands  |

## System Users

---

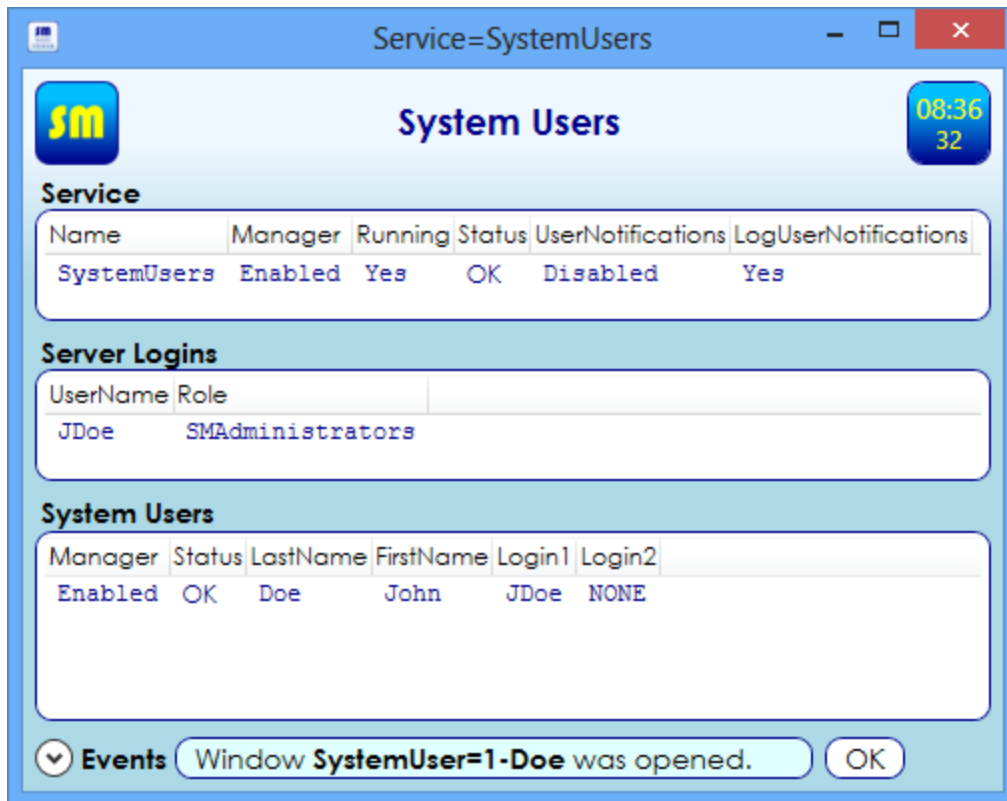
This service is responsible for management of system user list, user related security definitions, assignment of system event user notifications, and definition of system accessibility via SMWebLink to the various users via a web browser.

### System User List

---

To create a new system user record, proceed as follows:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click anywhere in the *System Users* display
- Select *New System User...*
- Enter user's last name
- Enter user's first name
- Enter user's account number. If there is no account number, enter 0
- Select *Save + Exit*



To configure user's parameters:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on user in the *System Users* display
- Change parameters as needed
- Select *Save + Exit*

## Security Credentials

Access to SM, whether via SM UI (Viewers), or a web browser (SMWebLink), is granted only to users with appropriate credentials. These are SQL server logins which determine level of access to SM for a user associated with the login. Each login, when created through SystemUsers service, consists of three parts: user name, password, and a role. The latter determines level of access to SM resources. The following roles are available:

|                  |  |
|------------------|--|
| SMAdministrators | Unrestricted access to SM resources  |
| SMOperators      | Users are permitted to view displays, execute commands (such as operate a control or poll an RTU), but are not permitted to change system configuration in any way       |
| SMMontitors      | Users are permitted to view displays, but are not permitted to execute commands (such as operate a control or poll an RTU), or to change system configuration in any way |

---

|             |  |
|-------------|--|
| SMCustomers | Reserved for future use, do not use at this time |
|-------------|--|

To create new login proceed as follows:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click anywhere in the *Server Logins* display
- Select *New Server Login...*
- Enter *login name* and press OK
- Enter *password* and press OK
- Select a *role*

One can also create a login imported from Windows OS on the server. In this case one simply imports a credential defined in Windows running on the server. Please note that this works only for server based accounts. To import a windows login:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click anywhere in the *Server Logins* display
- Select *import Server Login...*
- Select an existing Windows account
- Select a *role*

Each user can be associated with up to two logins. To associate a user with a login proceed as follows:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on the user of interest and select *Configuration...*
- Click on *Login1* (or *Login2*), and select one of the previously created server logins
- Select *Save + Exit*

Note, a user may have more than one login, but it is not advisable to give the same login to multiple users.

## Notifications

---

To enable notifications dispatching system-wide, proceed as follows:

- Start the *Main Viewer*
- Right-Click on *SystemUsers* service
- Set *UserNotifications* parameter to *True*
- Select *Save + Exit*

To configure a user to receive notifications the following must be configured:

- There must be at least one e-mail or SMS address associated with the user (see below)
- If the user is to receive all notifications, user's *Notifications* parameter must be set to *AllObjects*
- If the user is to receive only notification from certain I/O points
  - Those points must be added to user's ownership list (see below)
  - User's *Notifications* parameter must be set to *OwnedObjects*

To add I/O point to user's ownership list:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on the user of interest and select *Open*
- Right-click anywhere in the *I/O Points* display
- Select I/O point previously created via *RemoteStations* service

To remove I/O point from user's ownership list:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on the user of interest and select *Open*
- Right-click on I/O point of interest in the *I/O Points* display
- Select *Remove I/O Point* command

Note that removing I/O point from user's ownership list does not delete I/O point. Command only disassociates I/O point from the user.

To add an e-mail or SMS addresses to a user proceed as follows:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on the user of interest and select *Open*
- Right-click anywhere in the *Notifications Addresses* display and select *New Notification Address...*
- Enter the new e-mail address and press OK

If above e-mail address is to be an SMS address (see below):

- Right-click on the address in the *Notifications Addresses* display and select *Configuration...*
- Change *Type* parameter to *SMS*
- Select *Save + Exit*

SMS addresses are e-mail addresses specified here <http://www.emailtextmessages.com/> - for example, if SMS messages are to be sent to phone number (123) 456 7890 via Verizon, the following e-mail address must be entered into SM: 1234567890@vtext.com.

## SMWebLink

---

SM access via SMWebLink can be granted to a user as follows:

- Start the *Main Viewer*
- Open *SystemUsers* service window
- Right-click on the user of interest and select *Configuration...*
- Click on *Notifications* and select
  - *AllObjects* if user is to have access to all I/O points
  - *OwnedObjects* if user is to have access to owned I/O points only
  - *None* if user is to have no access to SM via SMWebLink
- Select *Save + Exit*

SMWebLink can be accessed at <https://www.btecentral.com> → SmartMaster → SMWebLink, or by simply going to <https://secure.btecentral.com/SmartMaster/WebLink/>. Please consult BTE when configuring firewall so SMWebLink is able to access your SCADA server.