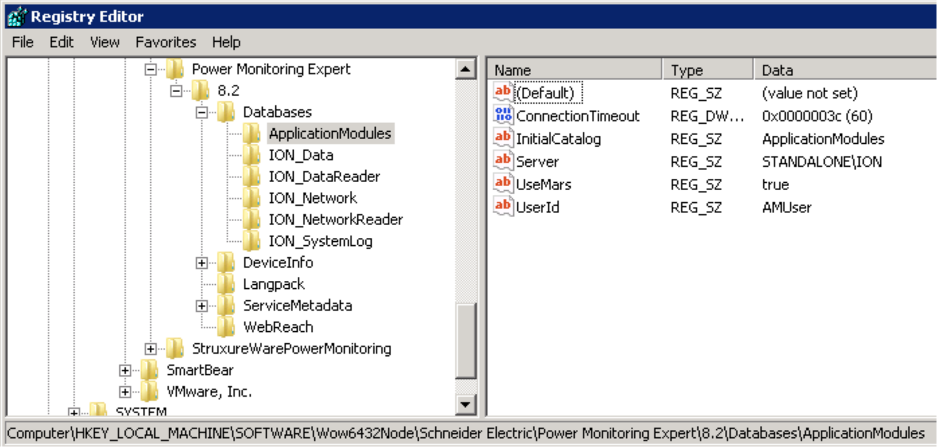
PME Connection Strings

For PME 8.2 connection string information has been consolidated into a single location, rather than keeping it spread out across the application in different locations such as database tables, registry keys and XML config files. All connection string information is now in the registry. This allows connection string information to be changed post install and stores the connection string information in the same place as the encrypted SQL account passwords.

During installation connection string entries are written to the registry under HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Schneider Electric\Power Monitoring Expert\8.2\Databases\.



- Power Monitoring Expert
  - 8.2
    - Databases
      - ApplicationModules
      - ION_Data
      - ION_DataReader
      - ION_Network
      - ION_NetworkReader
      - ION_SystemLog
    - DeviceInfo
    - Langpack
    - ServiceMetadata
    - WebReach
- StruxureWarePowerMonitoring
- SmartBear
- VMware, Inc.
- SYSTEM

Name	Type	Data
(Default)	REG_SZ	(value not set)
ConnectionTimeout	REG_DW...	0x0000003c (60)
InitialCatalog	REG_SZ	ApplicationModules
Server	REG_SZ	STANDALONE\ION
UseMars	REG_SZ	true
UserId	REG_SZ	AMUser

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Schneider Electric\Power Monitoring Expert\8.2\Databases\ApplicationModules

The database connection string information are contained in the keys ION_Data, ION_Network, ION_SystemLog and ApplicationModules. Four keys, one for each database. Each key has 4 values: Server, Initial Catalog, Timeout and UserId. The ApplicationModules key has an additional value 'UseMars'

	ApplicationModules	**ION_Data**	**ION_Network**	**ION_SystemLog**
Server	MachineName\ION	MachineName\ION	MachineName\ION	MachineName\ION
InitialCatalog	ApplicationModules	ION_Data	ION_Network	ION_SystemLog
UserId	AMUser	ION	ION	ION
ConnectionTimeout	60	60	60	60
UserMars	true	N/A	N/A	N/A

Based on the above table a connection string for ION_Data will be "server=MachineName\ION; database=ION_Data; User Id=ION;Password=UnencryptedPassword"

The are two other keys in the registry: ION_DataReader and ION_NetworkReader.

	ION_DataReader	**ION_NetworkReader**
MasterDb	ION_Data	ION_Network
UserId	ionedsd	ionedsd

UseMars	true	true

These keys override certain values of the ION_Data and ION_Network connection string information to create a connection string for the data source driver.

For ION_DataReader the connection string is "server=MachineName\ION; database=ION_Data; User Id = ionedsd; Password=UnencryptedPassword;MultipleActiveResultsSets=true"

The user id and the UseMars information is pulled from the ION_DataReader reg key while the rest of the connection string information is pulled from the ION_Data regkey.

Passwords

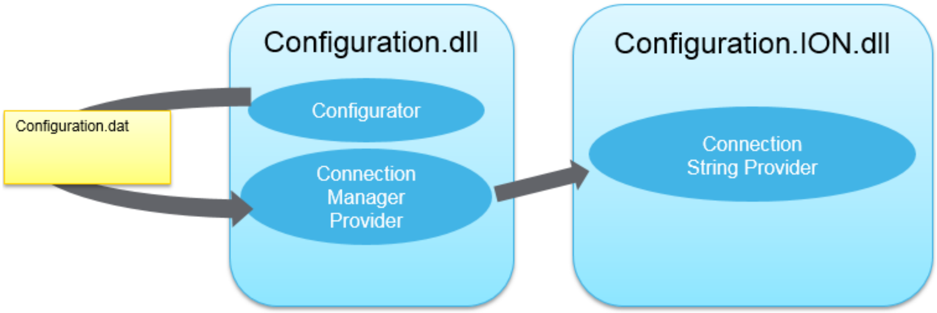
All SQL accounts share the same password stored encrypted at HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Schneider Electric\Power Monitoring Expert\8.2

UseMars

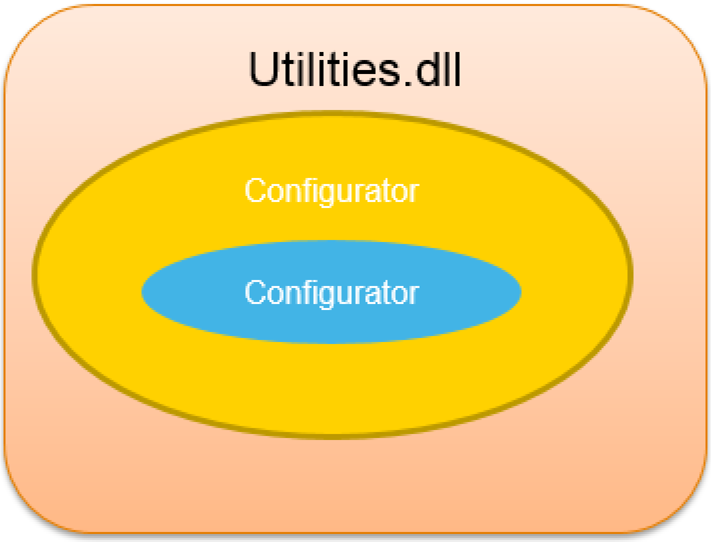
Mars stands for Multiple Active Results Sets. It is a feature of SQL Server to handle multiple active statements on a single connection. App mods code requires a connection string with 'MultipleActiveResultsSets=true'. Adding the registry key value pair UseMars:true, adds the MultipleActiveResultsSets=true to the resulting connection string.

Under the Hood: Configurator, Configurator and configuration.dat

The logic to parse the registry, build connection strings and decrypt in passwords lives in swcore. In a nutshell the Configurator class in Configuration.dll passes the Configuration.dat file into the ConnectionManagerProvider class which instantiates the provider mentioned in Configuration.dat. In practice that provider will always be the IONEnterprise.ConnectionStringProvider in Configuration.ION.dll.



Higher up in the architecture, in swione, you will see another Configurator class: PowerMeasurment.Utilities.Configurator. This class simply wraps the swcore version of Configurator and adds functionality to pull time outs from an app config.



Finally in application modules and the ION Data Source driver the intermediary steps of the Configurators and configuration.dat have been skipped. The ConnnectionStringProvider is directly instantiated from Configuration.ION.dll