

Document Name: pbsControl GSP Master OPC Server user manual

Version: 1.0.0

Date: Q4 2015

Developer: Kamjoo Bayat – Kamjoo.bayat@pbscontrol.com

Introduction

This document describes GSP Master OPC server configuration and using in SCADA systems.

User should know GSP protocol concepts for proper using of GSP OPC Server. For detail Information about GSP protocol please refer to GSP specification documentation.

http://pbscontrol.com/pdf/gsp_spec.pdf

We will talk about GSP protocol and concepts as it is possible for this document.

GSP (GPRS/3G for SCADA Projects) protocol is developed for SCADA systems which has following specification :

- There is not a stable communication media between RTUs and Master SCADA.
- RTUs has dynamic IP address.
- There is one Valid and static IP address for master SCADA.
- Time Synchronization is required between master SCADA and RTUS.
- Local Data buffering in RTU is required when Communication is offline and sending logged data to SCADA master after establishing connection.
- Number of total I/O for transition between RTU and Master SCADA is not more than 448 Tags
- There are many RTUs in Project. GSP is designed for SCADA with huge number of RTUs.
- Target project for GSP is power and gas distribution SCADA.
-

GSP client side is implemented in pbsSoftLogic (Linux Kernel) , icpDas UPac5000 with MiniOs7 and SSP GPRS RTU .

Installation

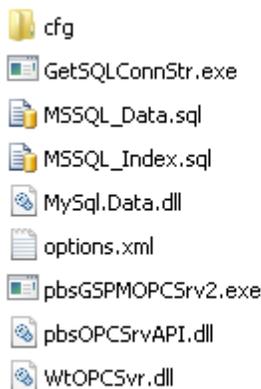
Download GSP Master OPC server from <http://www.pbscontrol.com/pbsProducts/pbsGSPMOPCSrv.zip>

Unzip pbsGSPMOPCSrv.zip at any directory on target system. As an example you will unzip it to c:\pbscontrol\GSPOPCsrv directory.

GSP Master OPC server is developed by C# and it needs following components on your system for proper running:

- Dot Net framework Runtime 3.5 SP1
<http://download.microsoft.com/download/2/0/E/20E90413-712F-438C-988E-FDAA79A8AC3D/dotnetfx35.exe>
-
- OPC DA2 .0 Core component distribution http://www.pbscontrol.com/runtime/OPCRT2_3.rar
run by administrator for proper OPC registration on computer.
- GSP OPC server is using MS SQL Server or MySQL for data archiving. Please check OPC server PC SQL Server and MySQL connection to Data base Server.
- RAM Disk Driver. GSP OPC server is using RAM Disk for communication with RTUS.
http://www.pbscontrol.com/runtime/Dataram_RAMDisk_V3.5.130R18.msi
- OPC Client /Browser utility.
http://www.pbscontrol.com/runtime/InstallOpcClassicDemoClient4_41_1.exe
-

After unzipping pbsGSPMOPCSrv.zip to c:\pbscontrol\GSPOPCsrv directory, you can see following files:



pbsGSPMOPCSrv2.exe: it is main OPC file. **For the first time run it as administrator for registering OPC server components in windows registry.**

WtOPCSvr.dll: OPC DA implementation by C.

pbsOPCSrvAPI.dll: Wrapper class for using WtOPCSvr.dll in Dot net CLR.

Options.xml: OPC configuration file

Cfg directory: RTU configuration directory .

MSSQL_Data.sql , MSSQL_Index.sql Microsoft SQL Server script for building index and data table .

GetSQLConnStr.exe simple utility to find SQL Server Connection String.

OPC configuration

Open options.xml file with any Editor. There are following parameters in this file:

```
<Node>
  <Name>MasterIPAddress</Name>
  <Desc>MasterIPAddress</Desc>
  <Value>10.71.3.50</Value>
</Node>
<Node>
  <Name>TCPIPPort</Name>
  <Desc>TCPIPPort</Desc>
  <Value>20000</Value>
</Node>
<Node>
  <Name>RTUOfflineTime</Name>
  <Desc>OfflineTime(sec)</Desc>
  <Value>300</Value>
</Node>
<Node>
  <Name>RAMDrivePath</Name>
  <Desc>RAMDrivePath</Desc>
  <Value>e:\SSPOPC</Value>
</Node>
<Node>
  <Name>SQLEnable</Name>
  <Desc>SQLLoggingEnable</Desc>
  <Value>0</Value>
</Node>
<Node>
  <Name>MySQLEnable</Name>
  <Desc>MySQLLoggingEnable</Desc>
  <Value>1</Value>
</Node>
<Node>
  <Name>SQLConnStr</Name>
  <Desc>SQLConnection String </Desc>
  <Value>server=localhost;user=root;database=pbshmi;port=3306;password=cccc;</Value>
</Node>
```

MasterIPAddress: IP address of SCADA I/O server. If there are more than one IP address on the I/O server, you need to set exact IP address which will communicate with RTUs, otherwise you will see error in Event viewer.

TCPIPPort : TCP port for communication with RTUs. All RTUs and Master SCADA should use same TCP port for proper communication. Please add TCP port to Server Firewall and allow both side communication.

RTUOfflineTime : OPC server will change RTU Online status to Offline , if there is no any valid communication between RTU and master SCADA for more than RTUOfflineTime Second .

RAMDrivePath : GSP OPC server is using RAM Drive for communication between RTUs and Master SCADA . Please download and install

http://www.pbscontrol.com/runtime/Dataram_RAMDisk_V3.5.130R18.msi to GSP OPC Server Computer. Set Drive Name in RAMDrivePath Parameter.

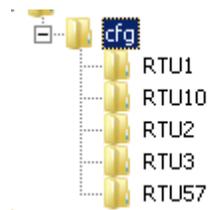
SQLEnable : Set this parameter to 1 if you want to log all RTU GSP data on MS SQL Server .

MySQLEnable : Set this parameter to 1 if you want to log all RTU GSP data on MySQL Server . you can use MS SQL Server or MySQL .

SQLConnStr : SQL or MySQL Connection string .

Cfg Directory:

You need to put RTU specifications in cfg directory. Make one directory for each RTU inside cfg directory and copy GSPTags.xml file there. For example if RTU id is 10 , make RTU10 inside cfg directory and copy GSPTags.xml file inside /cfg/RTU10 directory.



You can use any name for RTU directory. RTU ID is inside GSPTags.xml file.

GSPTags.xml file : you can see a simple configuration file as following :

```
<Version>1.0.0</Version>
<Params RTUID="10" Enable="True" pbsHMIPrefix="" />
<Tag Name="DoorState" Type="DI" Address="1" />
<Tag Name="TempState" Type="DI" Address="2" />
<Tag Name="PMStatus" Type="DI" Address="8" />

<Tag Name="THDVav" Type="AI" Address="1" />
<Tag Name="THDIav" Type="AI" Address="2" />
<Tag Name="H3V1" Type="AI" Address="3" />
<Tag Name="H5V1" Type="AI" Address="4" />
<Tag Name="H7V1" Type="AI" Address="5" />
<Tag Name="H3V2" Type="AI" Address="6" />
<Tag Name="H5V2" Type="AI" Address="7" />
<Tag Name="H7V2" Type="AI" Address="8" />
<Tag Name="H3V3" Type="AI" Address="9" />
<Tag Name="H5V3" Type="AI" Address="10" />
<Tag Name="H7V3" Type="AI" Address="11" />

<Tag Name="V12" Type="FI" Address="1" />
<Tag Name="V23" Type="FI" Address="2" />
<Tag Name="V31" Type="FI" Address="3" />
<Tag Name="I1" Type="FI" Address="4" />
<Tag Name="I2" Type="FI" Address="5" />
<Tag Name="I3" Type="FI" Address="6" />
<Tag Name="PSum" Type="FI" Address="7" />
<Tag Name="QSum" Type="FI" Address="8" />
<Tag Name="SSum" Type="FI" Address="9" />
<Tag Name="PfSum" Type="FI" Address="10" />
<Tag Name="Vunb" Type="FI" Address="11" />
<Tag Name="Iunb" Type="FI" Address="12" />

<Tag Name="TTTotal" Type="ULI" Address="1" />
<Tag Name="RETotal" Type="ULI" Address="2" />
```

Params.RTUID each RTU should has unique ID. ID is between 1 and 65535. There are 2 bytes for RTU ID in GSP Protocol.

Params.Enable Set it False if RTU is not enable in Project.

Params.pbsHMIPrefix if you are using pbsHMI as Master SCADA Software , then it is advised to use Data logging facility of GSP OPC server and only use pbsHMI for SCADA . normally pbsHMI added some prefix to Name of Tags and you need to set this prefix in this parameter . When you are using GSP OPC server for data logging, then you have 100% data backfilling facility with pbsHMI Platform.

You need to add one Tag Elements in GSPTags.xml file for each RTU GSP Tag.

Tag.Name : Name of Tag . It should be unique for each RTU.

Tag.Type : Tag Type . There are following Data Types in GSP Protocol:

- DI – Digital Input
- DO – Digital Output
- DOS – Digital Output Status
- AI- Analog Input (signed short)
- AO - Analog Output (signed short)
- AOS- Analog Output Status
- FI – Float input
- LI – Long Input (signed long)
- ULI – Unsigned Long Input

DO and AO will use for sending Command to RTU . DOS and AOS will use for reading DO and AO status.

Tag.Address : Tag address start from 1 and ended to 64 for each data type . so you can read maximum $7 \times 64 = 448$ Tags from RTU and write $2 \times 64 = 128$ Tags to RTU .

GSP frame will read all tags from RTU in one frame and Write DO and AO one by one to RTU by separate frames.

GSP OPC Runtime

Before running GSP OPC server you need to check following points in I/O Server computer:

- OPC runtime is installed properly.
- Dot Net 3.5 is installed with latest Service pack.
- You have proper connection with SQL server or MySQL if you want to use data logging.
- You configure options.xml file and cfg directory.

Run pbsGSPMOPCSrv2.exe as administrator for first time. GSP OPC server needs to do some modification in Windows Registry, so you should run it as administrator.

GSP OPC server will move to task bar automatically at startup .

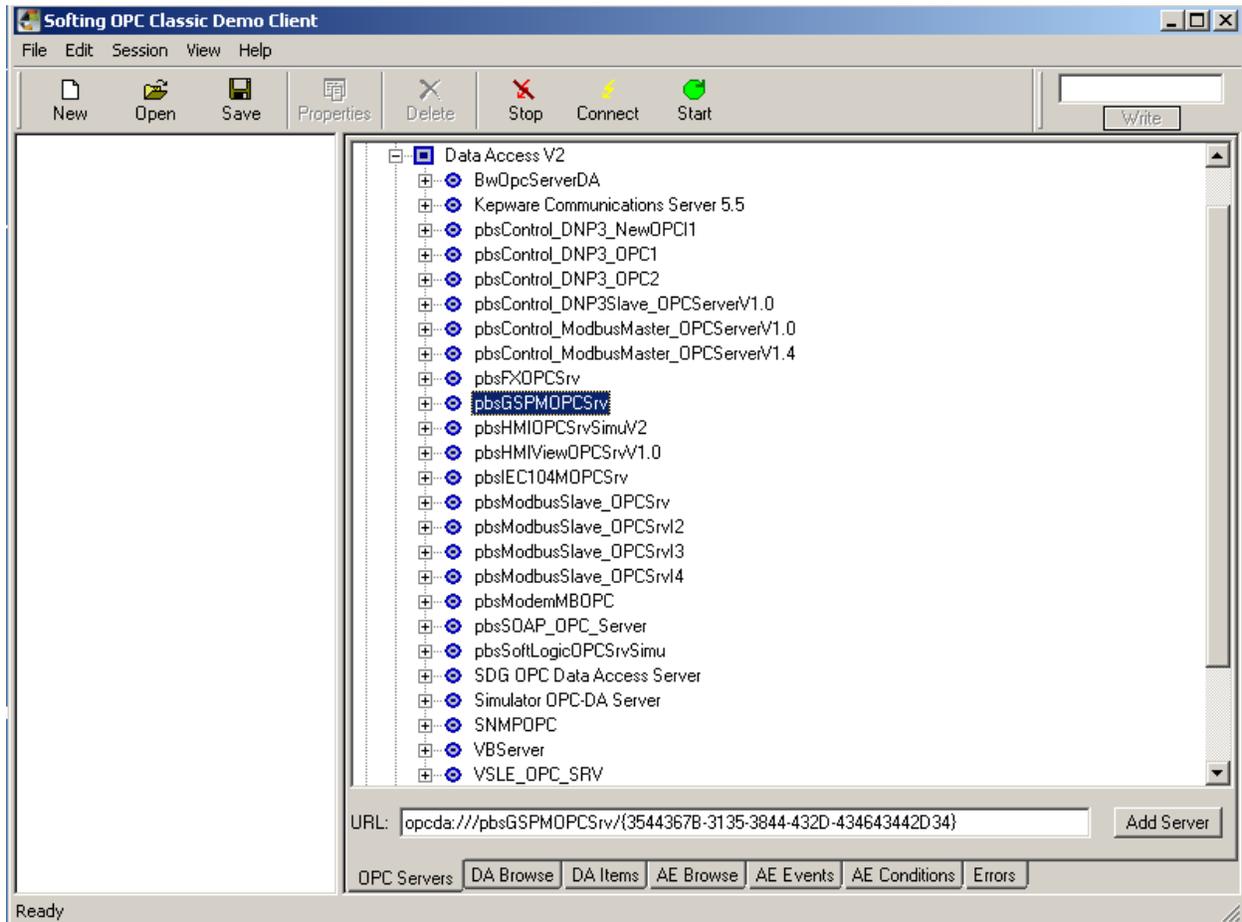


With double click on GSP OPC Server icon, you can see following form:



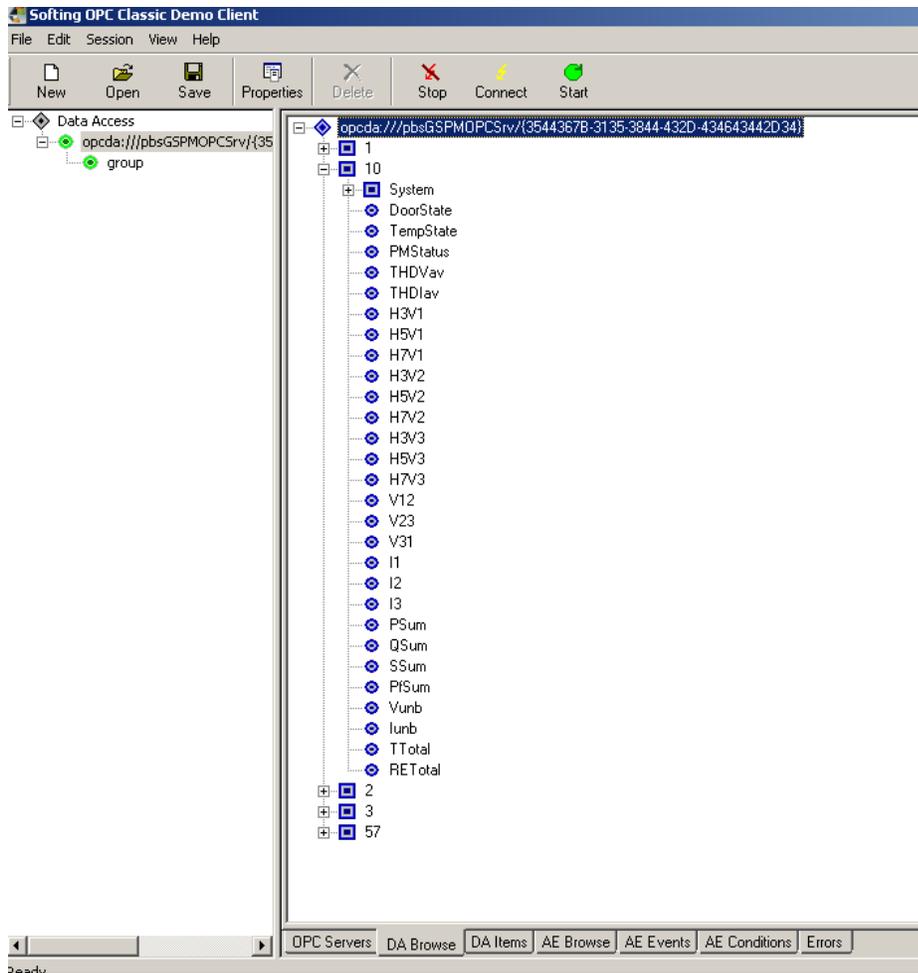
You can shut down OPC here and see OPC Start Time. Other OPC parameters can be read/write by any OPC browser.

Run Softing OPC Browser utility (or any other OPC browser utility) to see operation of GSP OPC server.



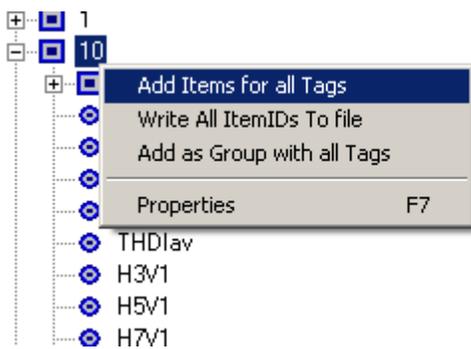
Open Data Access2 and then select pbsGSPMOPCSrv Item .

Double click on pbsGSPMOPCSrv for connection with OPC Server .



In DA Browser Tab you can see all defined RTU in the cfg directory.

Select one RTU and add all RTU tags by right click menu .



10.System.RecFrames	0	GOOD	12:04:38.692
10.System.NextScanForced	0	GOOD	12:04:38.692
10.System.NextScanSec	0	GOOD	12:04:38.692
10.DoorState	0	GOOD	12:04:38.692
10.TempState	0	GOOD	12:04:38.692
10.PMStatus	0	GOOD	12:04:38.692
10.THDVav	0	GOOD	12:04:38.692
10.THDIav	0	GOOD	12:04:38.692
10.H3V1	0	GOOD	12:04:38.692
10.H5V1	0	GOOD	12:04:38.692
10.H7V1	0	GOOD	12:04:38.692
10.H3V2	0	GOOD	12:04:38.692
10.H5V2	0	GOOD	12:04:38.692
10.H7V2	0	GOOD	12:04:38.692
10.H3V3	0	GOOD	12:04:38.692
10.H5V3	0	GOOD	12:04:38.692
10.H7V3	0	GOOD	12:04:38.692
10.V12	0	GOOD	12:04:38.692
10.V23	0	GOOD	12:04:38.692
10.V31	0	GOOD	12:04:38.692
10.I1	0	GOOD	12:04:38.692
10.I2	0	GOOD	12:04:38.692
10.I3	0	GOOD	12:04:38.692
10.PSum	0	GOOD	12:04:38.692
10.QSum	0	GOOD	12:04:38.692
10.SSum	0	GOOD	12:04:38.692
10.PfSum	0	GOOD	12:04:38.692
10.Vunb	0	GOOD	12:04:38.692
10.Iunb	0	GOOD	12:04:38.692
10.TTotal	0	GOOD	12:04:38.692
10.RETotal	0	GOOD	12:04:38.692
1.System.Online	1	GOOD	13:02:38.242
1.System.RecFrames	88	GOOD	13:02:38.242
1.System.NextScanForced	0	GOOD	12:04:38.680
1.System.NextScanSec	0	GOOD	12:04:38.680

RTU Tags in GSP OPC Server:

{RTUID}.{TagName} for user tags

{RTUID}.System.Online for showing Online status of RTU

{RTUID}.System.RecFrames for showing Number of Received frames from RTU . It will be reset when RTU going to offline.

{RTUID}.System.NextScanSec : It will send Next scan time to RTU .

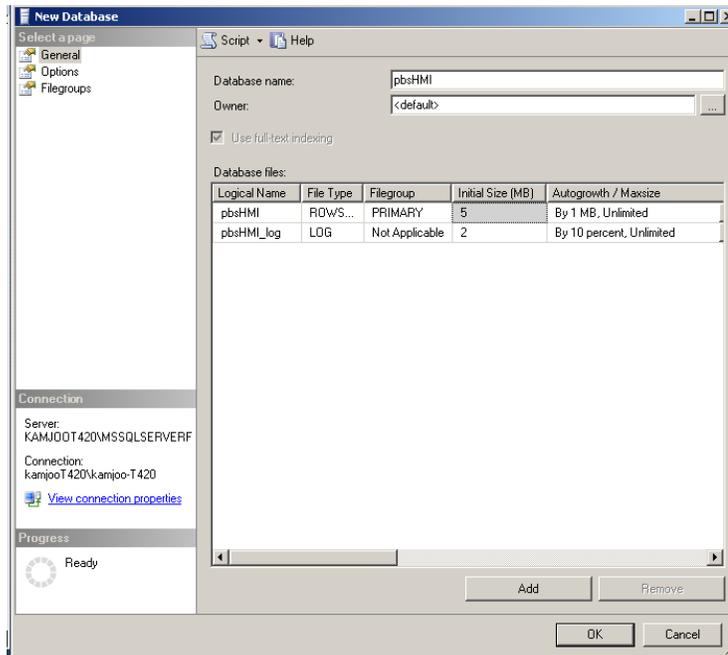
{RTUID}.System.NextScanForced : It will forced RTU to use New NextScanSec Parameter.

In above example RTU with ID 1 is online and it received 88 Frames.

GSP OPC Data Archiving on SQL Server

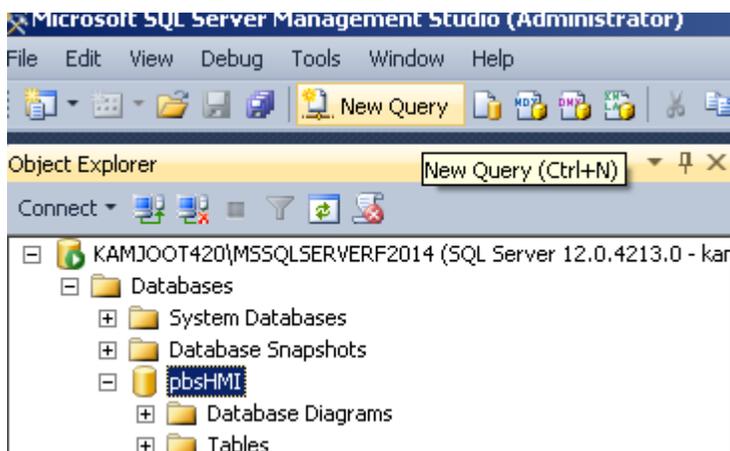
GSP OPC Server can archive data on MS SQL Server or MYSQL Server. In this part we will talk about MS SQL Server configuration.

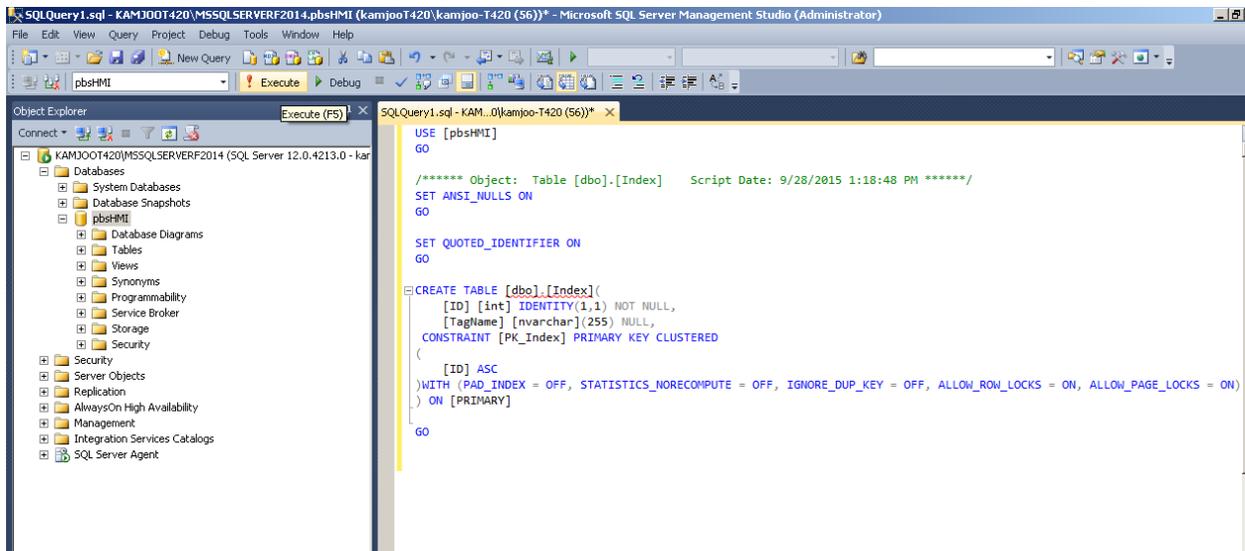
- 1- Make a new database in SQL Server with name pbsHMI.



Do not change other parameters and click on ok button.

- 2 – Make a new Query and copy / paste file content of MSSQL_Index.sql file in new query.



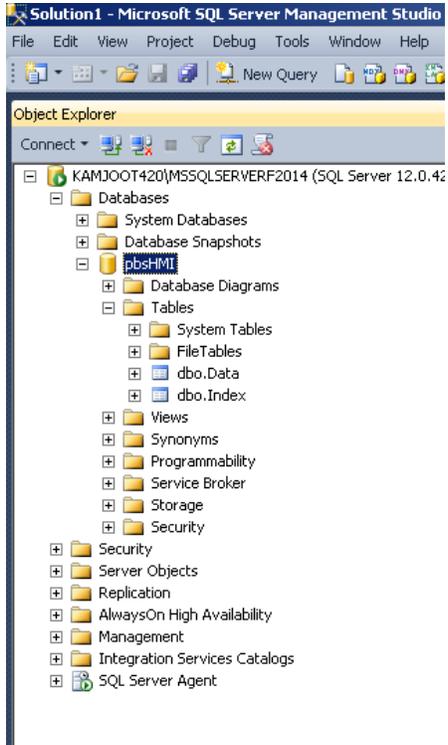


Click on **!execute** button.

This script will make index table in pbsHMI Database.

Repeat above steps and make data table inside pbsHMI database.

Now you have index and data table in pbsHMI Database.



Index Table : Index table has following fields .

Column Name	Data Type	Allow Nulls
ID	int	<input type="checkbox"/>
TagName	nvarchar(255)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

ID field is auto increment and will set by SQL Server.

When GSP OPC server wants to insert one tag to Database , First it will check Tag Name is exist in index table or not .if Tag Name is exist , then GSP OPC will get ID and Insert Data with Tag ID to Data table. Index table is not a big table and maximum it has all OPC tags.

Data Table :

Column Name	Data Type	Allow Nulls
ID	int	<input type="checkbox"/>
TagTime	bigint	<input type="checkbox"/>
TagValue	real	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Data Table is location for saving Tag Data . It has ID , TagTime and TagVale Fields .

TagTime is windows FileTime format .

Scripts for converting DateTime and FileTime:

`DateTimeToFileTime()` Converts the value of the current [DateTime](#) object to a Windows file time.

`DateTime.FromFileTime(long ft)` Converts the specified Windows file time to an equivalent local time.

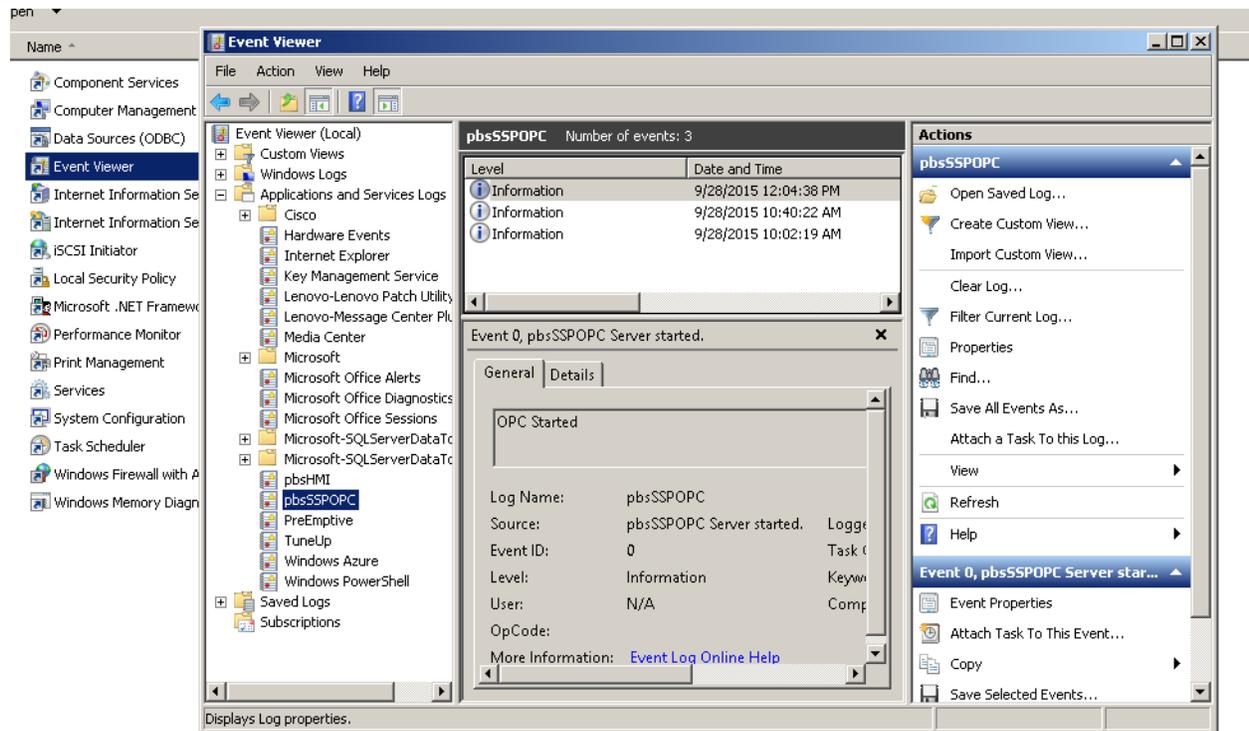
A *file time* is a 64-bit value that represents the number of 100-nanosecond intervals that have elapsed since 12:00 A.M. January 1, 1601 Coordinated Universal Time (UTC). The system records file times when applications create, access, and write to files.

GSP OPC Data Archiving on MySQL

GSP OPC Server Error Logging

GSP OPC server is using windows Event Viewer utility to log all systematic errors.

Run Event Viewer from Control panel /Administrative tools /Event Viewer



Open allocation and Service Logs and open pbsSSPOPC group .

In the right side panel you can see all internal errors and messages for GSP OPC Server .

When you have any problem with GSP OPC Server you can see source of problem and save logged error and send for maintenance to www.pbscontrol.com