GSP
GPRS for SCADA Projects Specification

www.pbscontrol.com

Kamjoo Bayat

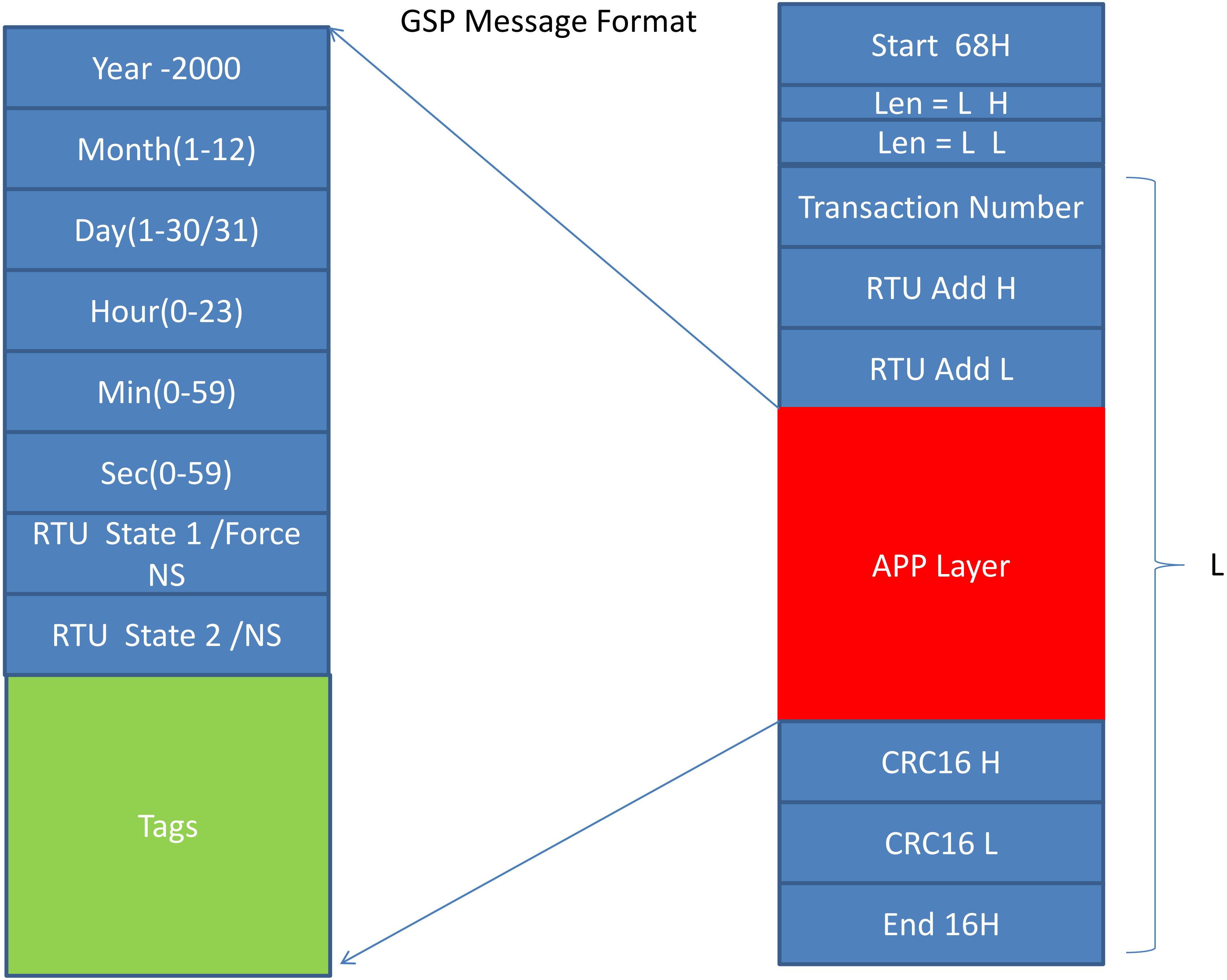
Ver 1.0.0

September 2015

Basic Concepts

- SCADA protocols like DNP3 and IEC870-5-104 are not working fine on GPRS /3G network because these protocols need permanent connection between Master SCADA and RTU with fix IP address for each RTU .
- GSP protocol is designed for optimum performance operation of Huge number of RTUs on a normal GPRS /3G network .
- GSP Project target is for power and Gas Distribution with limited number of I/O at each RTU and huge number of RTU .
- GSP is open standard from pbsControl and can be implemented for any type of low profile RTU with limited recourses and CPU power .
- Sample implementation of GSP client is developed by ANSI C and can be used free by any SCADA supplier .
- No need for static IP address for RTUs .
- We only need one fix and Valid IP Address for SCADA master .
- This protocol works on TCP connection between RTU and Master SCADA
- All Communications always starting from RTU
- Master Never start any transactions
- Protocol is binary based
- When RTU send a frame to Master , Master will answer within same session and send Master Time , Next scan time and DO or AO commands to RTU
- TCP connection will open for each transaction and will close when master answered
- There is an OPC Server In master SCADA for integration with any SCADA Platform .
- GSP Tag Address start from 1 for each Data Type. Address 0 is not valid .
- GSP Master OPC server is archived all GSP data on MS SQL Server .GSP message with old Time label are only archived in SQL Server and not passing to OPC Client.
- Master will synchronized RTU time at each communication with RTU .
- RTU is not permanently connected to Master SCADA . At each transaction , TCP connection will drop for optimum usage of GPRS/3G network .
- Each GSP message started with 0x68H and ended with 0x16H .

GSP Message Format



Start 68H

Len = L H

Len = L L

Transaction Number

RTU Add H

RTU Add L

APP Layer

CRC16 H

CRC16 L

End 16H

L

Year -2000

Month(1-12)

Day(1-30/31)

Hour(0-23)

Min(0-59)

Sec(0-59)

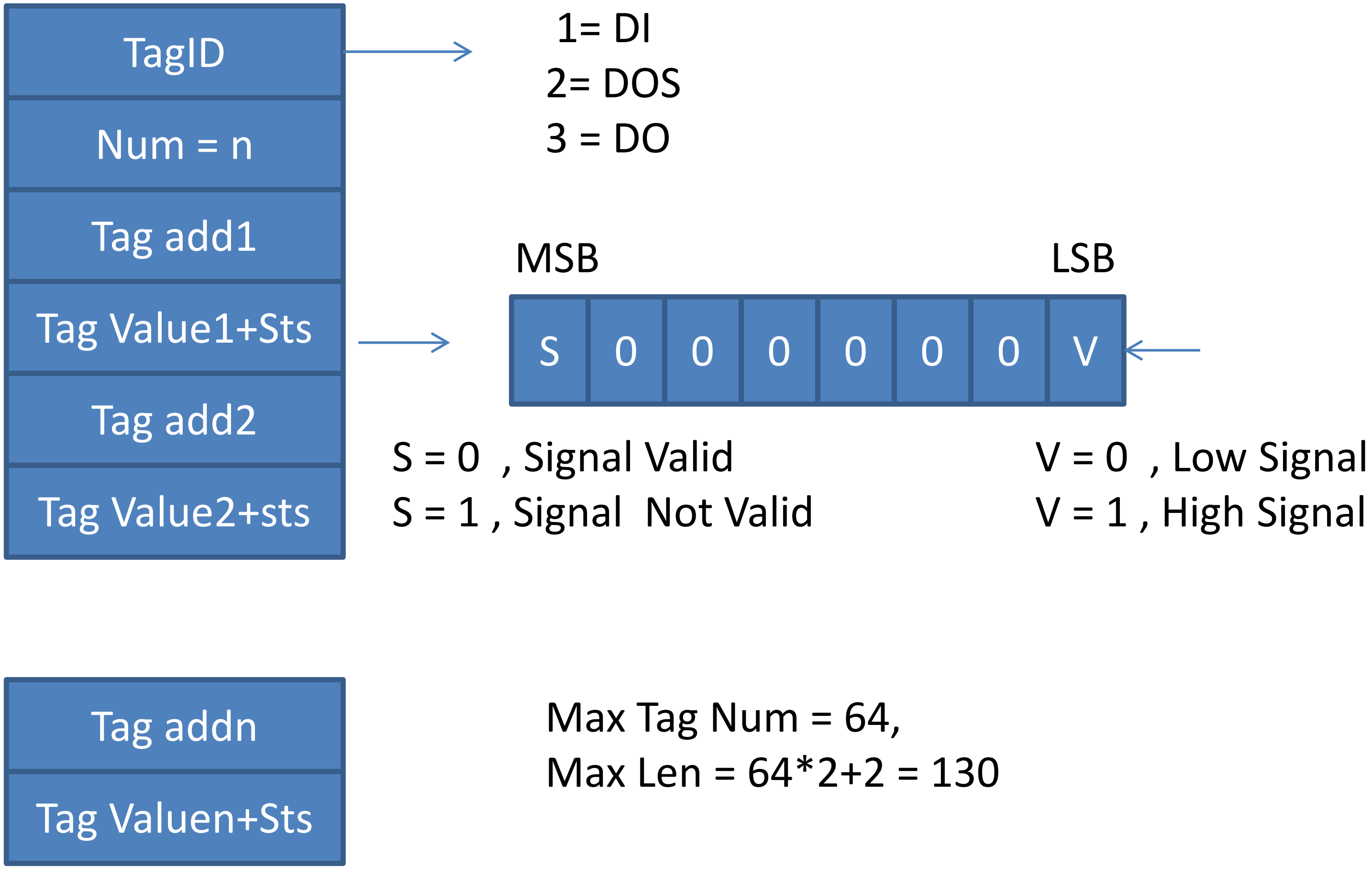
RTU State 1 /Force NS

RTU State 2 /NS

Tags

Data Types

- 1 = DI - Digital input
- 2 = DOS – Digital Output Status
- 3 = DO - Digital Output
- 4 = AI - Analog Input
- 5 = AOS - Analog Output Status
- 6 = AO - Analog output
- 7 = FI - Float Point
- 8 = LI - Long Integer
- 9 = ULI - Unsigned Long Integer



TagID

Num = n

Tag add1

Tag Value1+Sts

Tag add2

Tag Value2+sts

Tag addn

Tag Valuen+Sts

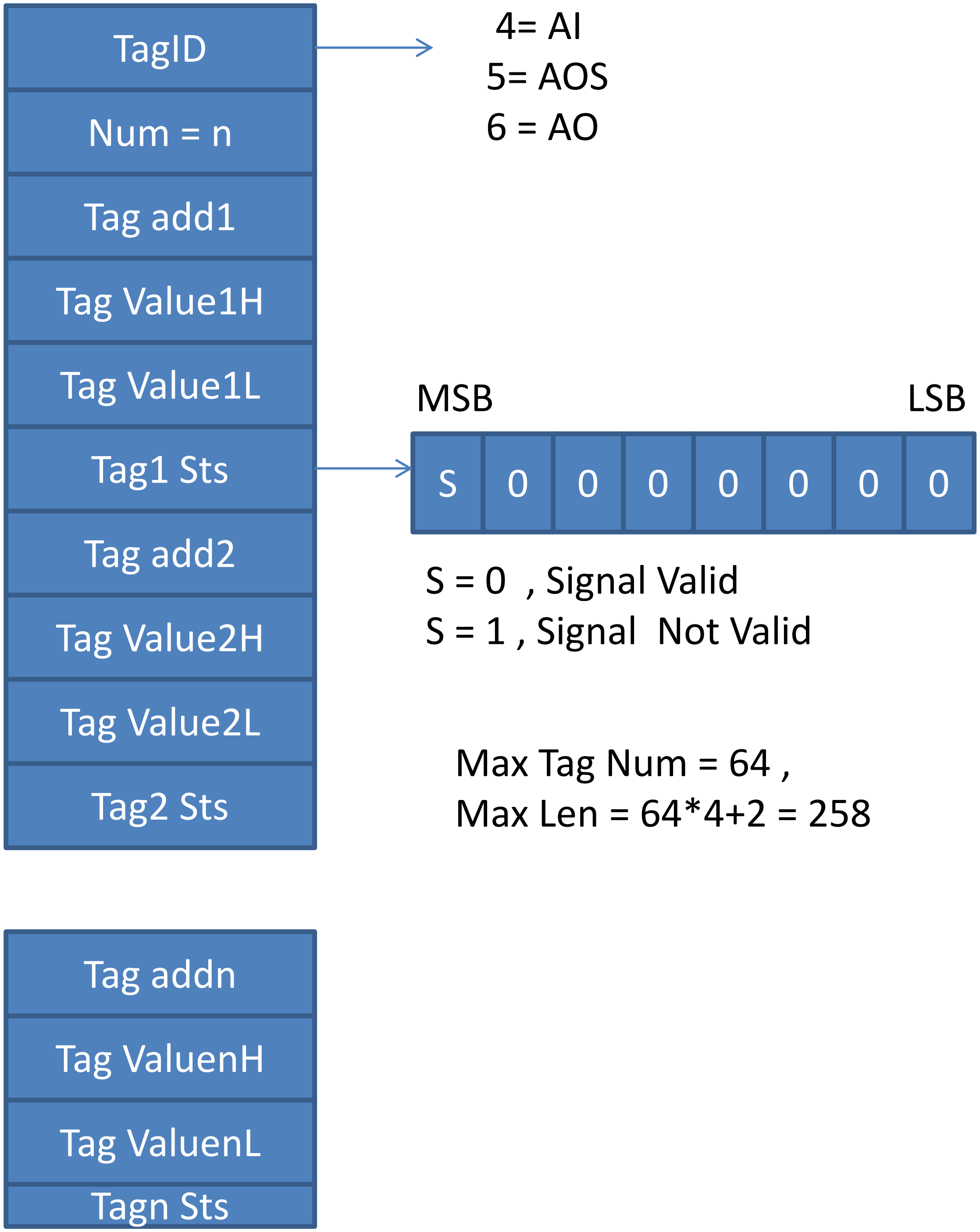
1= DI
2= DOS
3 = DO

MSB							LSB
S	0	0	0	0	0	0	V

S = 0 , Signal Valid
S = 1 , Signal Not Valid

V = 0 , Low Signal
V = 1 , High Signal

Max Tag Num = 64,
Max Len = 64*2+2 = 130



TagID

Num = n

Tag add1

Tag Value1H

Tag Value1L

Tag1 Sts

Tag add2

Tag Value2H

Tag Value2L

Tag2 Sts

Tag addn

Tag ValuenH

Tag ValuenL

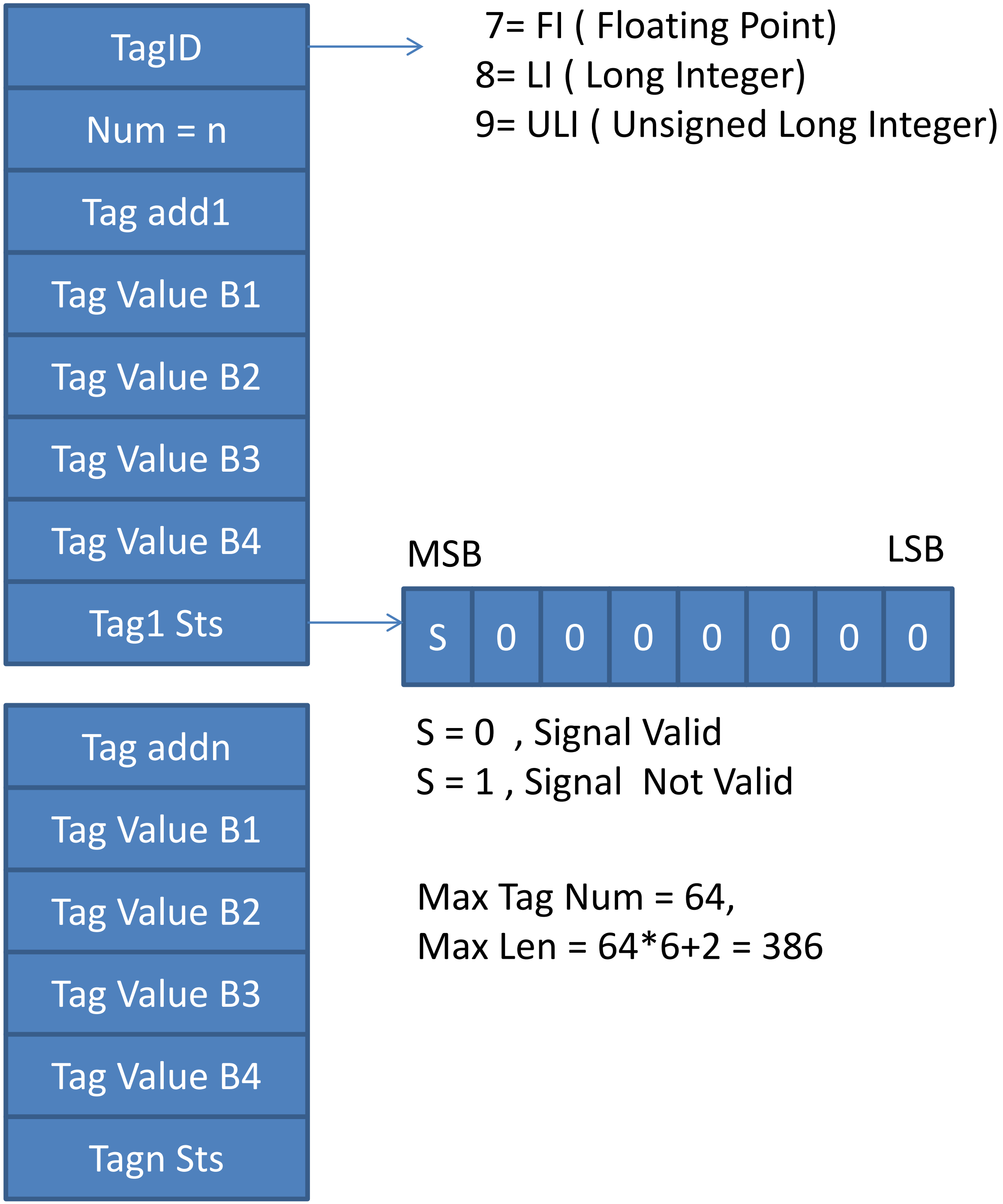
Tagn Sts

4= AI
5= AOS
6 = AO

MSB							LSB
S	0	0	0	0	0	0	0

S = 0 , Signal Valid
S = 1 , Signal Not Valid

Max Tag Num = 64 ,
Max Len = 64*4+2 = 258



TagID

Num = n

Tag add1

Tag Value B1

Tag Value B2

Tag Value B3

Tag Value B4

Tag1 Sts

Tag addn

Tag Value B1

Tag Value B2

Tag Value B3

Tag Value B4

Tagn Sts

7= FI (Floating Point)
8= LI (Long Integer)
9= ULI (Unsigned Long Integer)

MSB							LSB
S	0	0	0	0	0	0	0

S = 0 , Signal Valid
S = 1 , Signal Not Valid

Max Tag Num = 64,
Max Len = 64*6+2 = 386

Send Frame from RTU to Master

- RTU State 1 , 2 are RTU health status
  - – Status1 , Bit3 = 1 , RTU Restarted and need time

Send Frame from Master to RTU

- RTU State 1 = Forced by Master for next scan
  - 1 = Forced
  - 0= user default value for sending next frame
- RTU State 2 = Next scan Time (Sec)
- RTU State1.T when Write by Master to RTU with value 1 , shows it has only Time Synch Command and do not changed Scan force values

MSB							LSB
0	0	0	0	0	T	D	F

F = 0 , user Default Scan Time

F = 1 , Forced next scan time

D = 0 , There is no tags In Frame

D = 1 , There are tags in frame

T = 0 , RTU Normal Operation

T = 1 , RTU Restarted and need Time

Transaction Number

- Start from 0 and increase for each send frame by RTU
- Master will return same value in response
- Maximum transaction number is 255 and it will reset to 0 by RTU .
- When GSP master is getting GSP messages with not sequenced Transaction number , it will find some messages are archived locally In RTU and couldn’t transfer to Master .

Notes

- Total Frame Length is maximum 1634 bytes
- RTU can send different type of tags in one frame
- DI , AI ,AOS , FI , LI , ULI (AO Status) ,DOS (DO status) are send from RTU to Master
- DO , AO are send from Master to RTU
- Protocol can transfer maximum 64 Tags for each type (Total 64 *7 = 448 Tags Read and 2x64 = 128 Tags Write)
- When RTU couldn't transfer GSP message to master , it will save GSP message in a local file . File name is getting from Current Date Time
- When Communication is stablished with master SCADA , RTU sends all logged GSP messages to master and will delete saved GSP file from local drive.

GSP Operation

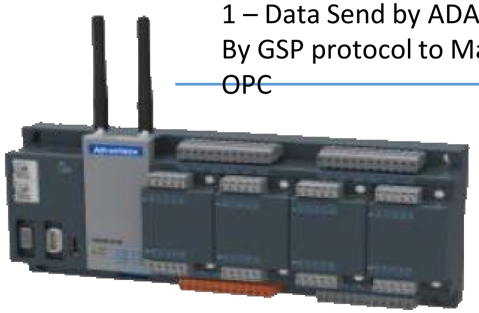
3 – GSP Master OPC Server check is there any command for RTU
If yes send command on same session with Current Master Time to RTU
If there is no command only send current master Time for time sync to RTU

4– RTU check is there any command in the frame , if yes update output tags
If not sync RTU time with master
5 – disconnect connection for next transition
6 – if GSP frame is for logged file (when RTU is offline) , delete log file .

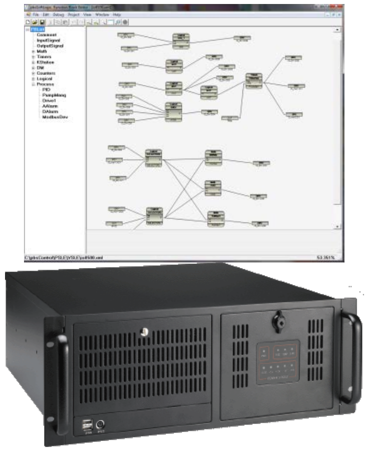
0- Connect to master SCADA
1 – Data Send by ADAM-3600
By GSP protocol to Master
OPC

IRAN CELL , HAMRAHE AVL ,
RITEL
GSP Protocol Over
GPRS/3G Network

2 – GSP Master OPC get frame and
Save data into SQL Server by frame time
And update OPC tags



ADAM-3600
8 AI
8 DI
4 DO
Any type of SIM Card with
enabled GPRS/3G
Dynamic IP address



Master SCADA
GSP OPC Server
MS SQL Server
pbsHMI
Fix and valid IP address

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

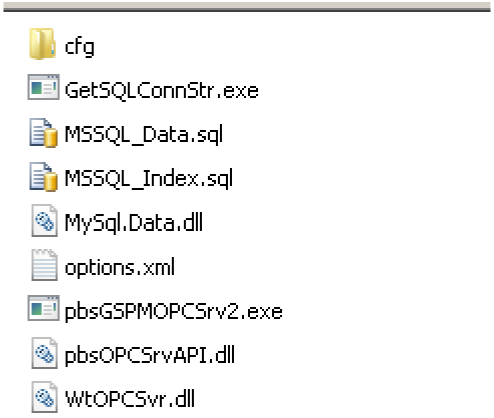
Download GSP Master OPC server from http://www.pbscontrol.com/pbsProducts/pbsGSPMOPCSrv.zip

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:



cfg
GetSQLConnStr.exe
MSSQL_Data.sql
MSSQL_Index.sql
MySql.Data.dll
options.xml
pbsGSPMOPCSrv2.exe
pbsOPCSrvAPI.dll
WtOPCSvr.dll

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

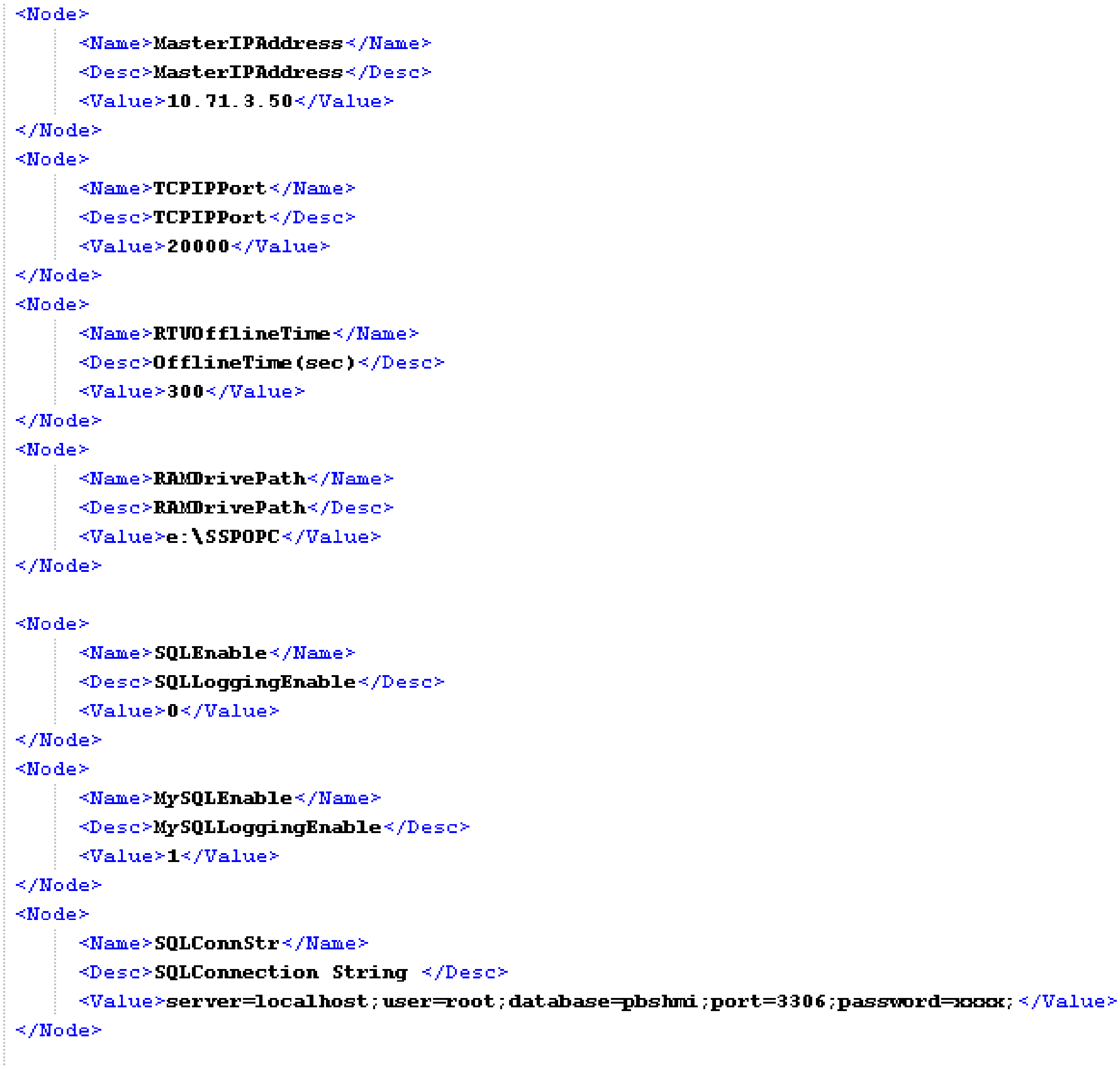
Cfg directory: RTU configuration directory .

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=xxxx; </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 be 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.

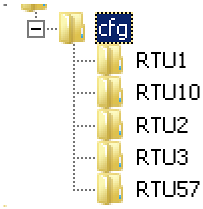
SQLEnable : Set this parameter to 1 if you want to log all RTU GSP data on MS SQL Server .

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

SQLConnStr : SQL or MySQL Connection string .

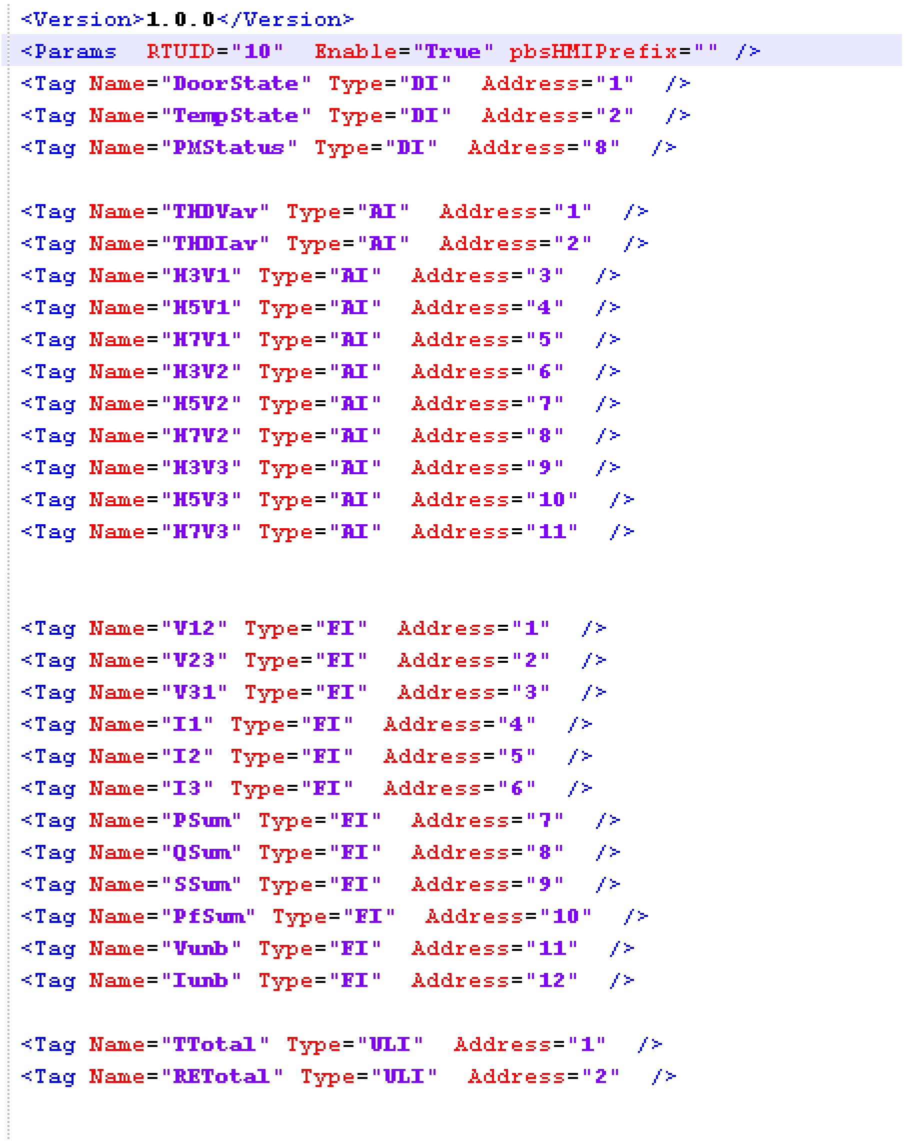
Cfg Directory:

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="TTotal" 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 7x64 = 448 Tags from RTU and write 2x64 = 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.

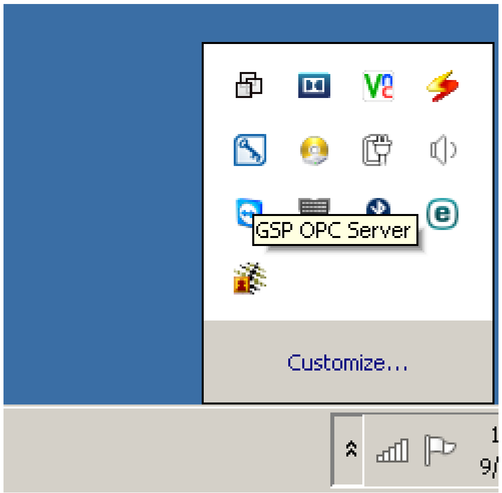
GSP OPC Runtime

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

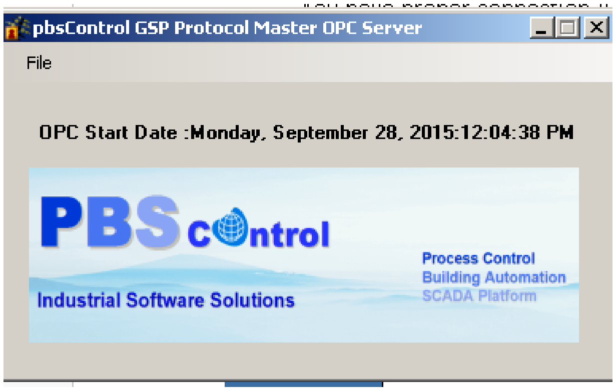
- OPC runtime is installed properly.
- Dot Net 3.5 is installed with latest Service pack.
- You have proper connection with SQL server or MySQL if you want to use data logging.
- You configure options.xml file and cfg directory.

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 .



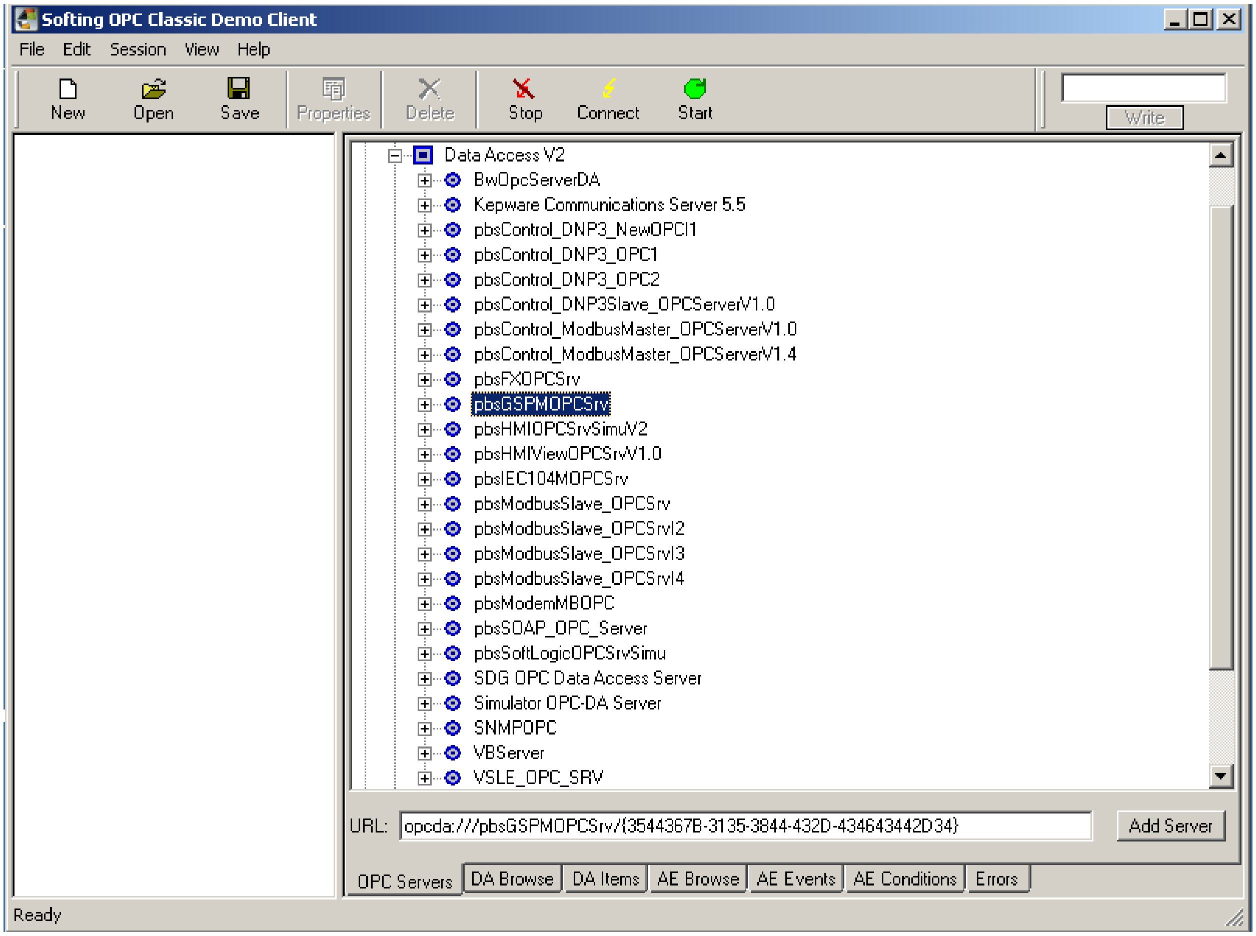
With double click on GSP OPC Server icon, you can see following form:


pbsControl GSP Protocol Master OPC Server
File
PBS c ntrol
Process Control
Building Automation
SCADA Platform
Industrial Software Solutions

OPC Start Date :Monday, September 28, 2015:12:04:38 PM

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.

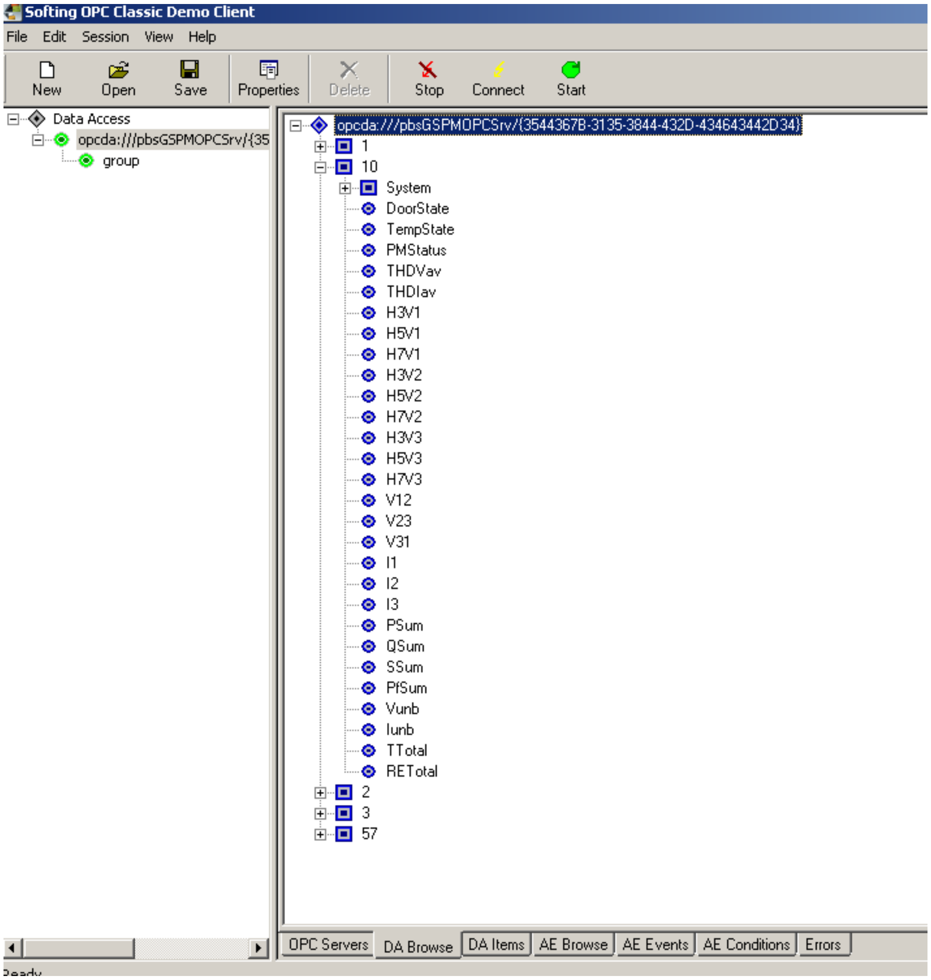


Data Access V2
- BwOpcServerDA
- Kepware Communications Server 5.5
- pbsControl_DNP3_NewOPCI1
- pbsControl_DNP3_OPC1
- pbsControl_DNP3_OPC2
- pbsControl_DNP3Slave_OPCServerV1.0
- pbsControl_ModbusMaster_OPCServerV1.0
- pbsControl_ModbusMaster_OPCServerV1.4
- pbsFXOPCSrv
- pbsGSPMOPCSrv
- pbsHMIOPCSrvSimuV2
- pbsHMIViewOPCSrvV1.0
- pbsIEC104MOPCSrv
- pbsModbusSlave_OPCSrv
- pbsModbusSlave_OPCSrvl2
- pbsModbusSlave_OPCSrvl3
- pbsModbusSlave_OPCSrvl4
- pbsModemMBOPC
- pbsSOAP_OPC_Server
- pbsSoftLogicOPCSrvSimu
- SDG OPC Data Access Server
- Simulator OPC-DA Server
- SNMPOPC
- VBServer
- VSLE_OPC_SRV

URL: opcda:///pbsGSPMOPCSrv/{3544367B-3135-3844-432D-434643442D34}

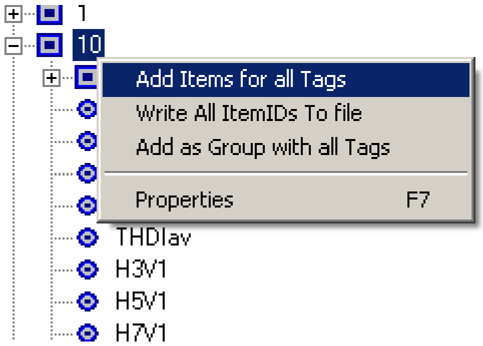
Open Data Access2 and then select pbsGSPMOPCSrv Item .

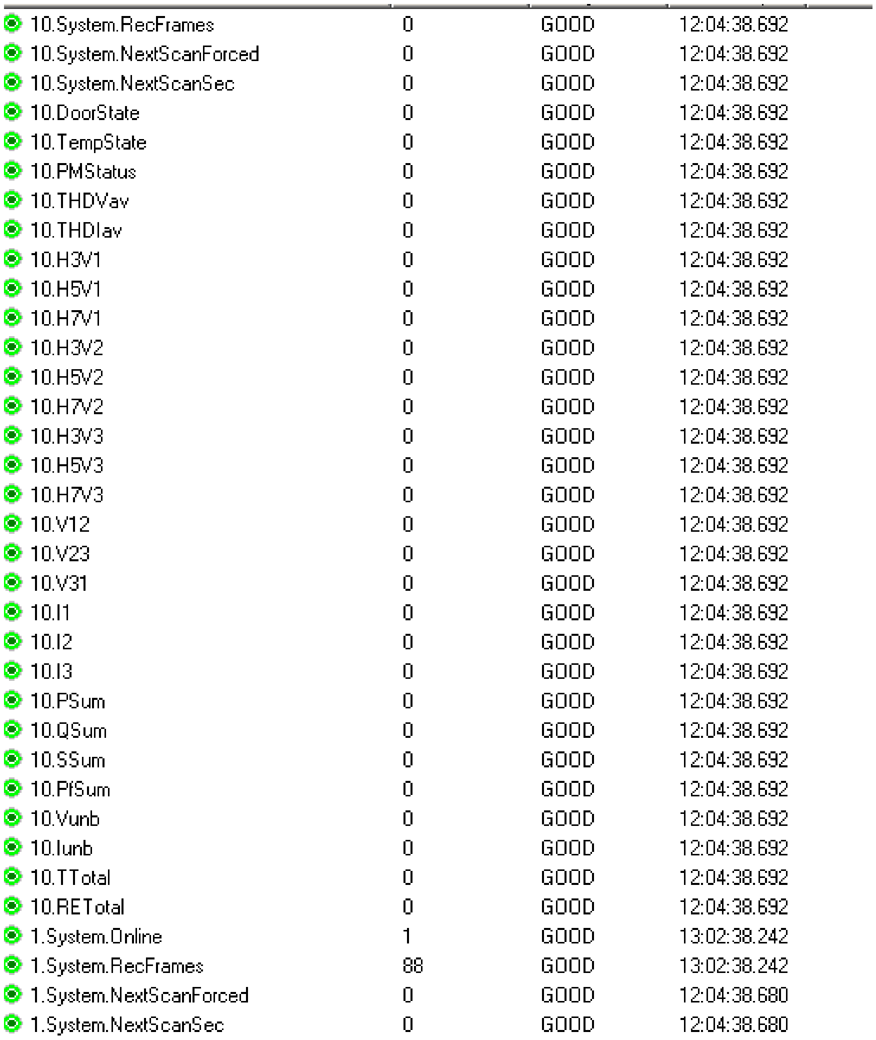
Double click on pbsGSPMOPCSrv for connection with OPC Server .



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 .

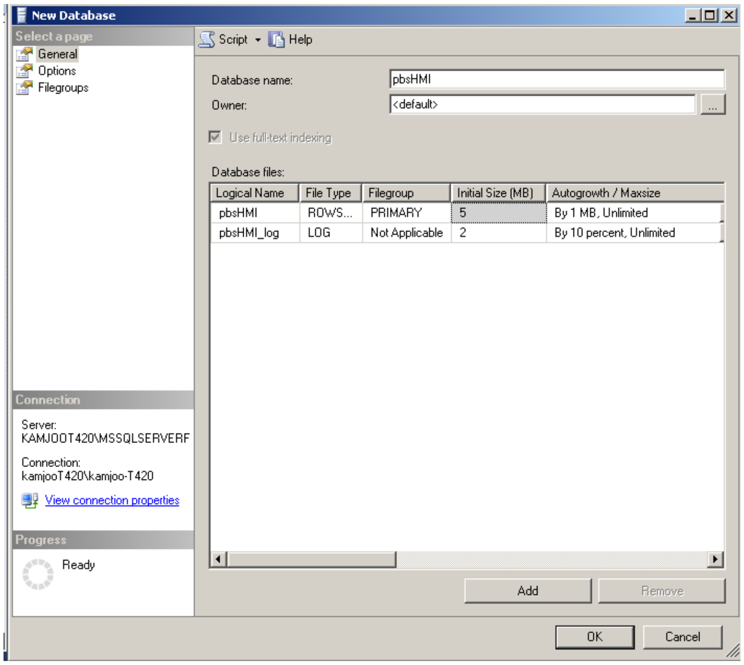
{RTUID}.System.NextScanForced : It will forced RTU to use New NextScanSec Parameter.

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.



Database name: pbsHMI

Owner: <default>

☑ Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize
pbsHMI	ROWS...	PRIMARY	5	By 1 MB, Unlimited
pbsHMI_log	LOG	Not Applicable	2	By 10 percent, Unlimited

Server:
KAMJOOT420\MSSQLSERVERF

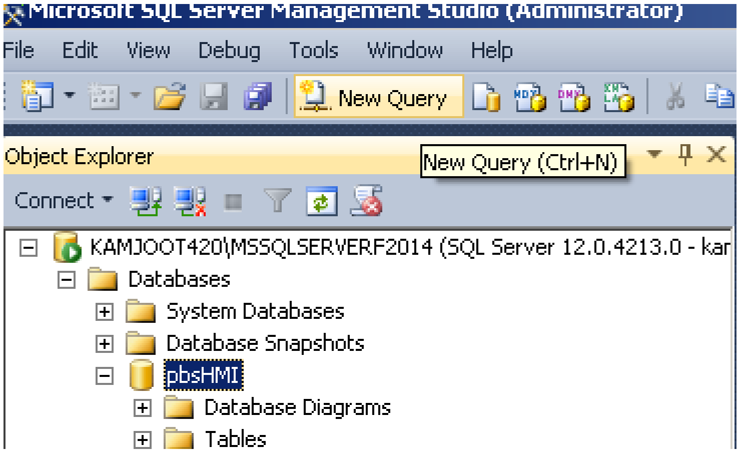
Connection:
kamjooT420\kamjoo-T420

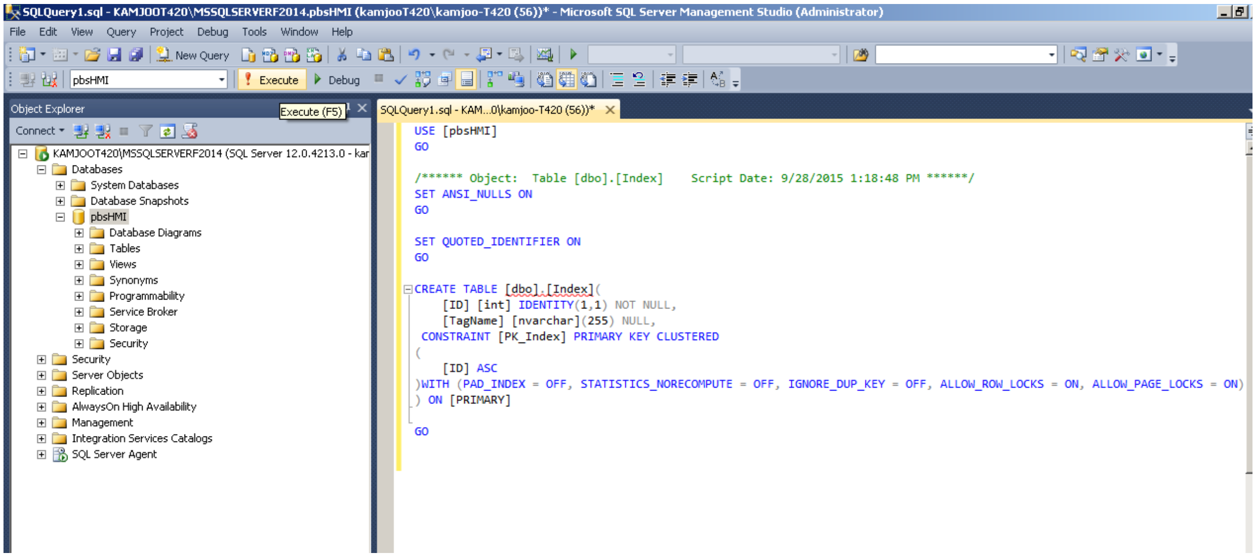
View connection properties

Ready

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.




SQLQuery1.sql - KAMJOOT420\MSSQLSERVERF2014.pbsHMI (kamjooT420\kamjoo-T420 (56))* - Microsoft SQL Server Management Studio (Administrator)
File Edit View Query Project Debug Tools Window Help
New Query
pbsHMI
Execute
Debug
Execute (F5)
Object Explorer
Connect
KAMJOOT420\MSSQLSERVERF2014 (SQL Server 12.0.4213.0 - kar
Databases
System Databases
Database Snapshots
pbsHMI
Database Diagrams
Tables
Views
Synonyms
Programmability
Service Broker
Storage
Security
Security
Server Objects
Replication
AlwaysOn High Availability
Management
Integration Services Catalogs
SQL Server Agent
SQLQuery1.sql - KAM...0\kamjoo-T420 (56))*

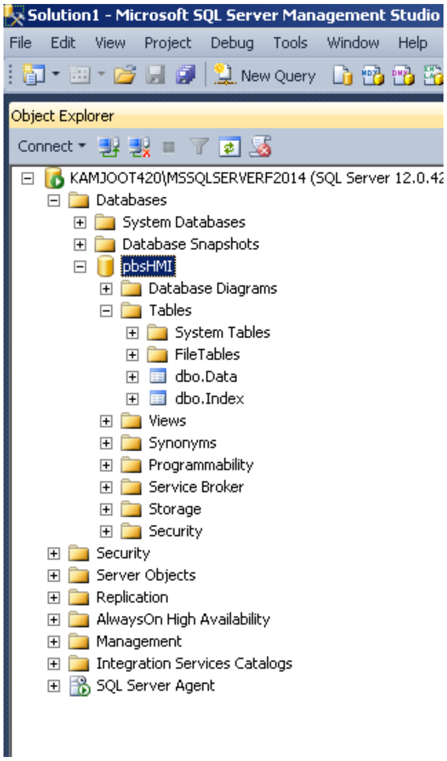
```
USE [pbsHMI]
GO

/****** Object:  Table [dbo].[Index]    Script Date: 9/28/2015 1:18:48 PM ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Index](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [TagName] [nvarchar](255) NULL,
 CONSTRAINT [PK_Index] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON)
) ON [PRIMARY]

GO
```

Click on !execute button.

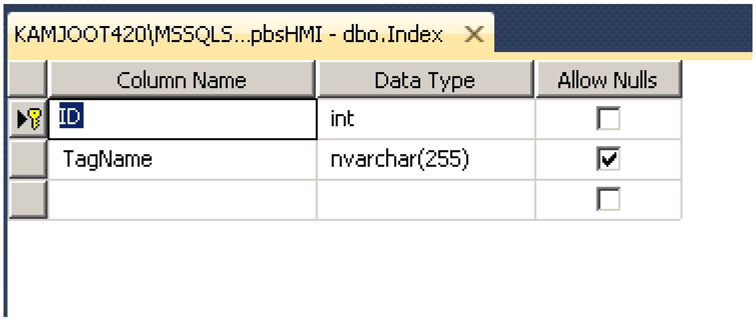
This script will make index table in pbsHMI Database.

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 .



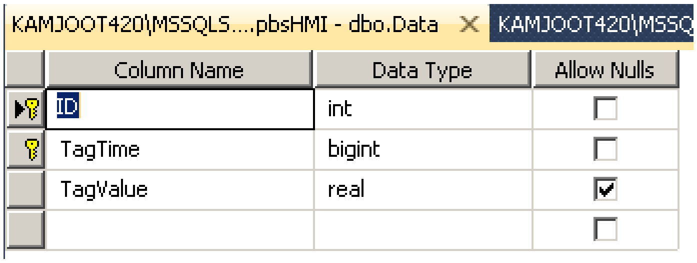
KAMJOOT420\MSSQLS...pbsHMI - dbo.Index

Column Name	Data Type	Allow Nulls
ID	int	□
TagName	nvarchar(255)	☑
		□

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 :



KAMJOOT420\MSSQLS....pbsHMI - dbo.Data KAMJOOT420\MSSQ

Column Name	Data Type	Allow Nulls
ID	int	□
TagTime	bigint	□
TagValue	real	☑
		□

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:

DataTime.ToFileTime() 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.

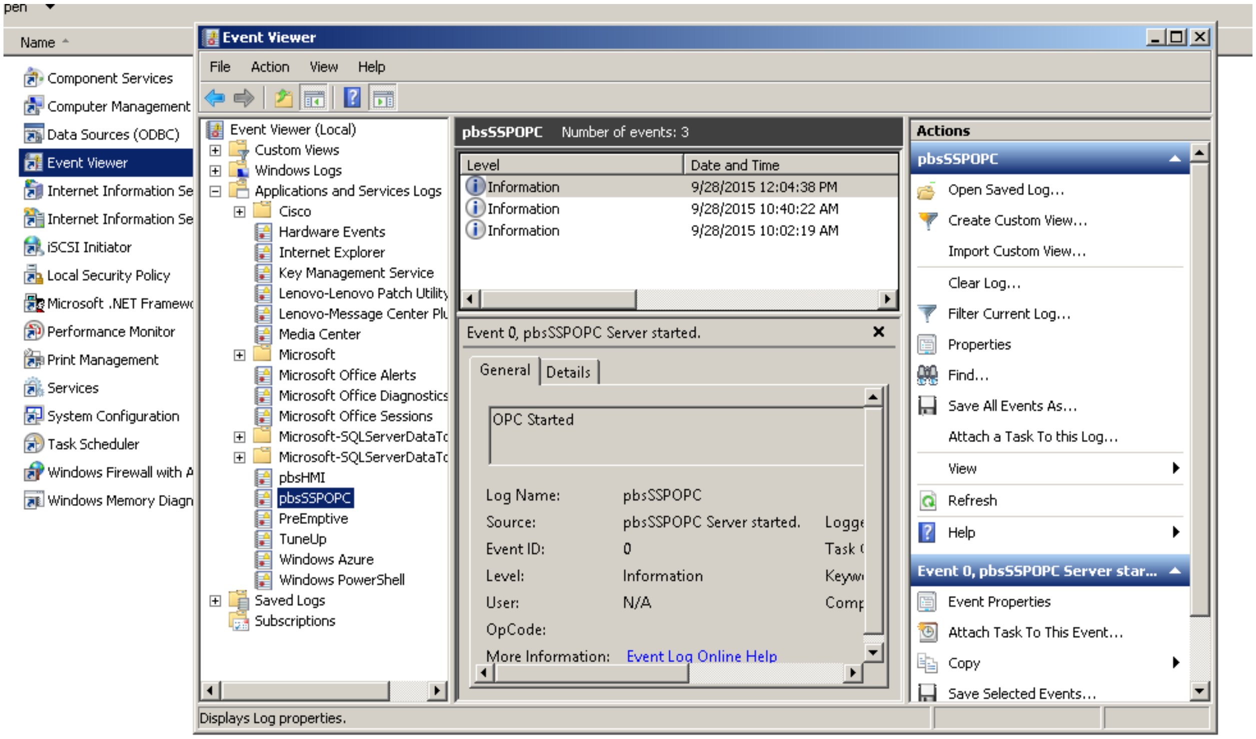
A *file time* is a 64-bit value that represents the number of 100-nanosecond intervals that have elapsed since 12:00 A.M. January 1, 1601 Coordinated Universal Time (UTC). The system records file times when applications create, access, and write to files.

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



pbsSSPOPC Number of events: 3

Level	Date and Time
Information	9/28/2015 12:04:38 PM
Information	9/28/2015 10:40:22 AM
Information	9/28/2015 10:02:19 AM

Event 0, pbsSSPOPC Server started.

OPC Started

Log Name: pbsSSPOPC
Source: pbsSSPOPC Server started. Logge
Event ID: 0 Task (
Level: Information Keyw
User: N/A Comp
OpCode:
More Information: Event Log Online Help

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