OPC Factory Server-Tuning the Communication Parameters

04/2014

EIO0000001730.01

www.schneider-electric.com

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

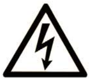
Table of Contents

Safety Information

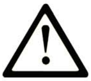
Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.



The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

About the Book

At a Glance

Document Scope

This document describes tuning of the OFS communication parameters.

Validity Note

The document has been updated with the release of OFS V3.50.

Related Documents

Title of Documentation	Reference Number
OPC Factory Server V3.50 User manual	35008244

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Product Related Information

Contents

Chapter 1
Tuning the Communication Channel in a Vijeo Citect Environment

Introduction

This chapter provides guidelines and practical tips to analyze the performance of an existing system, and then configure some of the OFS parameters to optimize the overall performance.

The steps given in this chapter have been explained with an example.

NOTE: This document targets only devices with Ethernet TCP/IP communication (configured with MBT address in OFS configuration tool).

What Is in This Chapter?

This chapter contains the following topics:

Before Starting

Updating the Target Ethernet Module

Before working on OFS, make sure that the communication module is configured.

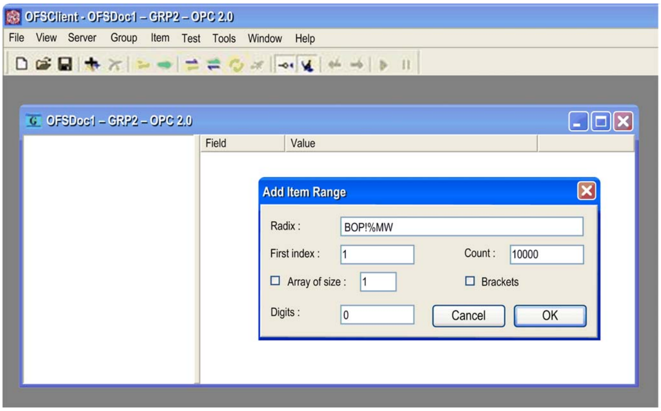
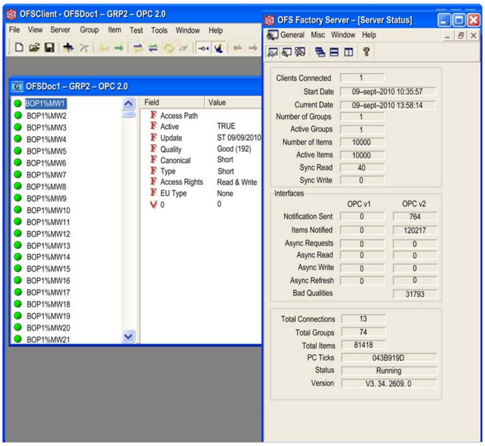
If you are using Quantum communication modules, make sure that the firmware of the CPU and the Ethernet module are updated to the latest version to improve the communication performance.

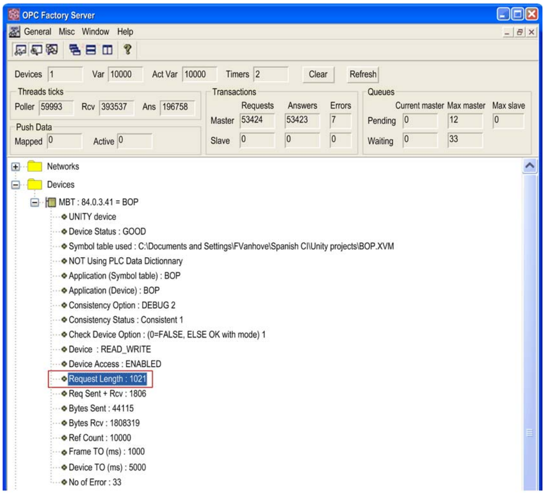
As an example, the latest release of 140NOE771•• module allows packets of 1024 bytes instead of 256 and also allows 12 concurrent requests instead of 4 compared to previous versions of CPU modules. Upgrading this firmware therefore, increases the theoretical Ethernet port throughput by 12 times. This works only with versions > V2.80 of the CPU module.

Testing Performance with the OFS Test Client

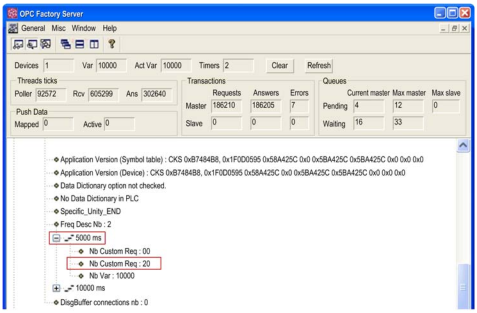
Determining the Number of Requests Required to Scan the PLC

To test the OFS request generation for determining the number of requests required to scan the PLC, perform the following steps:

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>1</td><td>Run the OFS test client and add the items needed in your real SCADA project in the <b>Add Item Range</b> dialog box.
<b>Result:</b>

Add Item Range
Radix : BOP!%MW
First index : 1
Count : 10000
☐ Array of size : 1
☐ Brackets
Digits : 0

OFSDoc1 – GRP2 – OPC 2.0
BOP1%MW1, BOP1%MW2, BOP1%MW3, BOP1%MW4, BOP1%MW5, BOP1%MW6, BOP1%MW7, BOP1%MW8, BOP1%MW9, BOP1%MW10, BOP1%MW11, BOP1%MW12, BOP1%MW13, BOP1%MW14, BOP1%MW15, BOP1%MW16, BOP1%MW17, BOP1%MW18, BOP1%MW19, BOP1%MW20, BOP1%MW21
Field / Value
Access Path
Active: TRUE
Update: ST 09/09/2010
Quality: Good (192)
Canonical: Short
Type: Short
Access Rights: Read &amp; Write
EU Type: None
0: 0
OFS Factory Server – [Server Status]
Clients Connected: 1
Start Date: 09–sept–2010 10:35:57
Current Date: 09–sept–2010 13:58:14
Number of Groups: 1
Active Groups: 1
Number of Items: 10000
Active Items: 10000
Sync Read: 40
Sync Write: 0
Interfaces (OPC v1 / OPC v2)
Notification Sent: 0 / 764
Items Notified: 0 / 120217
Async Requests: 0 / 0
Async Read: 0 / 0
Async Write: 0 / 0
Async Refresh: 0 / 0
Bad Qualities: 31793
Total Connections: 13
Total Groups: 74
Total Items: 81418
PC Ticks: 043B919D
Status: Running
Version: V3. 34. 2609. 0
In the example shown above, the SCADA project has 10,000 items, so <code>%MW1</code> to <code>%MW10000</code> is added in the OFS test client.</td></tr>
</table>

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>Configure the group with a slow update rate. For example, 5000 ms.</td></tr>
<tr><td>3</td><td>Open the network window in OFS.
<b>Result:</b>


OPC Factory Server
General Misc Window Help
Clear
Refresh

Devices 1 Var 10000 Act Var 10000 Timers 2
Threads ticks: Poller 59993 Rcv 393537 Ans 196758
Push Data: Mapped 0 Active 0
Transactions: Requests / Answers / Errors
Master 53424 53423 7
Slave 0 0 0
Queues: Current master / Max master / Max slave
Pending 0 12 0
Waiting 0 33
Networks
Devices
MBT : 84.0.3.41 = BOP
UNITY device
Device Status : GOOD
Symbol table used : C:\Documents and Settings\FVanhove\Spanish Cl\Unity projects\BOP.XVM
NOT Using PLC Data Dictionnary
Application (Symbol table) : BOP
Application (Device) : BOP
Consistency Option : DEBUG 2
Consistency Status : Consistent 1
Check Device Option : (0=FALSE, ELSE OK with mode) 1
Device : READ_WRITE
Device Access : ENABLED
Request Length : 1021
Req Sent + Rcv : 1806
Bytes Sent : 44115
Bytes Rcv : 1808319
Ref Count : 10000
Frame TO (ms) : 1000
Device TO (ms) : 5000
No of Error : 33


OPC Factory Server
General Misc Window Help
Clear
Refresh

Devices 1 Var 10000 Act Var 10000 Timers 2
Threads ticks: Poller 92572 Rcv 605299 Ans 302640
Push Data: Mapped 0 Active 0
Transactions: Requests / Answers / Errors
Master 186210 186205 7
Slave 0 0 0
Queues: Current master / Max master / Max slave
Pending 4 12 0
Waiting 16 33
Application Version (Symbol table) : CKS 0xB7484B8, 0x1F0D0595 0x58A425C 0x0 0x5BA425C 0x5BA425C 0x0 0x0 0x0
Application Version (Device) : CKS 0xB7484B8, 0x1F0D0595 0x58A425C 0x0 0x5BA425C 0x5BA425C 0x0 0x0 0x0
Data Dictionary option not checked.
No Data Dictionary in PLC
Specific_Unity_END
Freq Desc Nb : 2
5000 ms
Nb Custom Req : 00
Nb Custom Req : 20
Nb Var : 10000
10000 ms
DisgBuffer connections nb : 0
As shown in the example above, the network window displays the information about the <b>Request Length</b> supported by the target Ethernet port and the number of requests required to read all the subscribed OPC items (<b>Nb VarMan Req</b>).
This information enables you to verify that the communication port on the PLC is correctly configured and is also important to calculate the time required by OFS to refresh the entire OPC group containing your desired items.</td></tr>
</table>

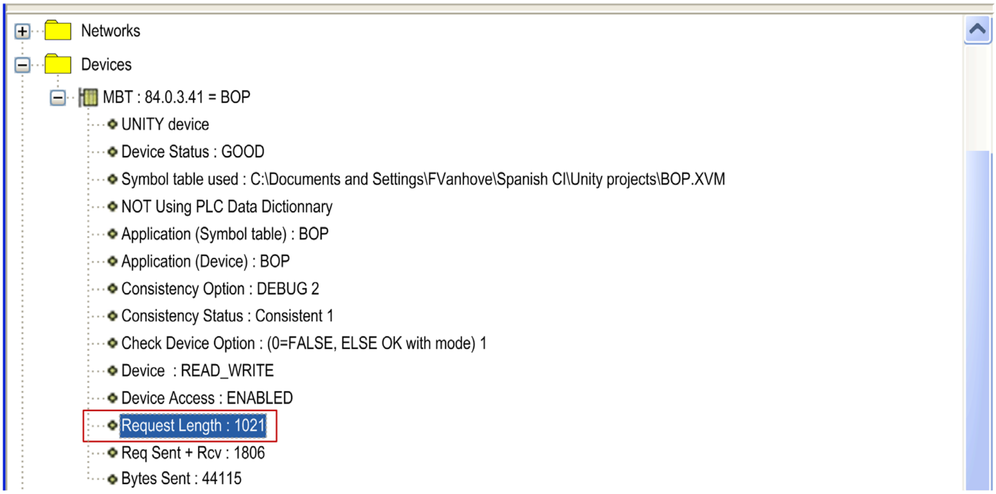
NOTE: In the real SCADA project, the **Nb VarMan Req** may be higher because the required addresses are not contiguous (in different memory block area) so OFS needs to send more requests.

Calculating the Time Needed to Scan the Entire Group

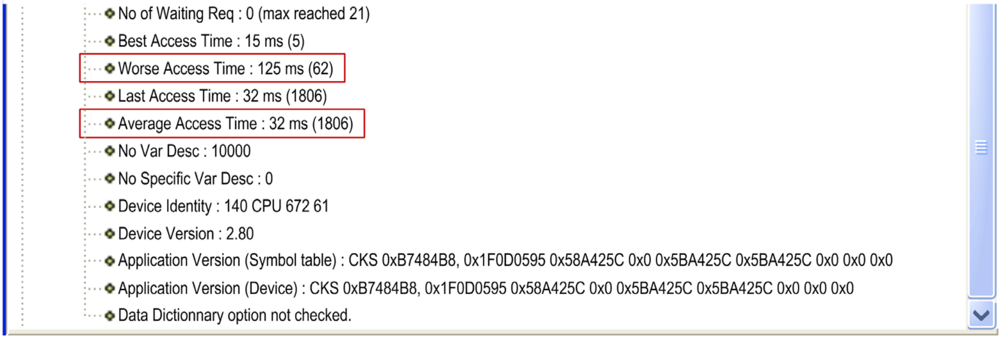
Run the OFC test client for a few minutes, and then check the **Average Access Time** in the OFS network window. Also, check the **Worse Access Time**, and the number of requests with **Worse Access Time** (shown between brackets). This information indicates the network health and stability. These access times are directly linked to the PLC task time, CPU load, and the network bandwidth.

As a general rule, you can evaluate the communication time for a single request using the formula, Request Time = Transmission Time Request + (2 x PLC Scan) + Transmission Time Answer.

To calculate the time needed to scan the entire group (that is, the time needed to scan all the active items in a given group once), you need to multiply the **Nb VarMan Req** by the **Average Access Time**.



```
Networks
Devices
  MBT : 84.0.3.41 = BOP
    UNITY device
    Device Status : GOOD
    Symbol table used : C:\Documents and Settings\FVanhove\Spanish CI\Unity projects\BOP.XVM
    NOT Using PLC Data Dictionnary
    Application (Symbol table) : BOP
    Application (Device) : BOP
    Consistency Option : DEBUG 2
    Consistency Status : Consistent 1
    Check Device Option : (0=FALSE, ELSE OK with mode) 1
    Device  : READ_WRITE
    Device Access : ENABLED
    Request Length : 1021
    Req Sent + Rcv : 1806
    Bytes Sent : 44115
```



```
    No of Waiting Req : 0 (max reached 21)
    Best Access Time : 15 ms (5)
    Worse Access Time : 125 ms (62)
    Last Access Time : 32 ms (1806)
    Average Access Time : 32 ms (1806)
    No Var Desc : 10000
    No Specific Var Desc : 0
    Device Identity : 140 CPU 672 61
    Device Version : 2.80
    Application Version (Symbol table) : CKS 0xB7484B8, 0x1F0D0595 0x58A425C 0x0 0x5BA425C 0x5BA425C 0x0 0x0 0x0
    Application Version (Device) : CKS 0xB7484B8, 0x1F0D0595 0x58A425C 0x0 0x5BA425C 0x5BA425C 0x0 0x0 0x0
    Data Dictionnary option not checked.
```

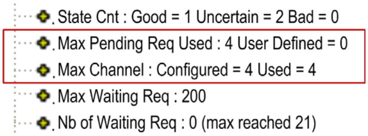
In the example above, the **Nb VarMan Req** is 20 and the **Average Access Time** is 32 ms; therefore the time needed to scan the group is 20*32 ms = 640 ms.

Since OFS can send multiple requests in parallel, you need to divide the above result by the number of parallel requests. The number of requests sent in parallel to the device depends on **Max Channel** and **Max Pending** parameters.

Max Channel is the number of channels (number of TCP/IP connections for MBT alias) available with the device. It can also be considered as the number of requests processed in parallel by the device in one PLC scan. The maximum number of connections that can be available to the device is device dependent. Refer to Estimation of Network Performance *(see OPC Factory Server V3.50, User Manual)*. **Max Pending** is the number of requests sent by the server to the device waiting for a response.

By default, in the OFS configuration tool, **Max Channel** is set to 4, and **Max Pending** is set to 0. **Max Pending** = 0 means that the user lets the OFS server identify the target communication port of the PLC and determine how many parallel requests can be sent at the same time. OFS bases its decision as per a predefined table listing the type and reference of the communication module (NOE/COPRO/ETY, and so on).

You can configure **Max Pending**, either at 0, or at the same value as **Max Channel**. You can check the number of **Max Pending** requests determined by OFS from the network window of OFS. You can also read this value directly from the OPC client: <alias name> `!#NbrMaxPending`. The actual time needed to scan the entire OPC group is therefore evaluated using the formula (**Nb VarMan Req** x **Average Access Time**) / **Max Pending Req Used**.



State Cnt : Good = 1 Uncertain = 2 Bad = 0
Max Pending Req Used : 4 User Defined = 0
Max Channel : Configured = 4 Used = 4
Max Waiting Req : 200
Nb of Waiting Req : 0 (max reached 21)

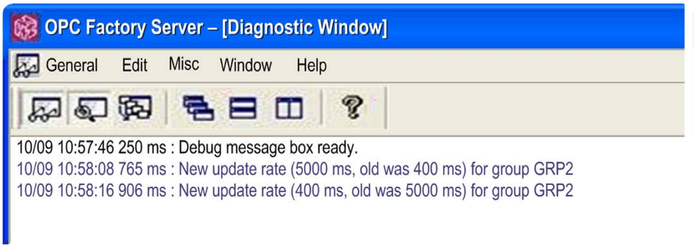
In the above example, the configured value of **Max Channel** is 4 and **Max Pending** is 0 (**Max Pending Req Used** is set by OFS to 4). If you want to maximize communication performance, you can configure **Max Channel** to 12. To avoid consuming all communication bandwidth available in PLC (in case Unity Pro is to be connected to PLC), you can retain **Max Channel** to 4. The time required to scan 10,000 items is (20 x 32)/4 = 160 ms.

NOTE: You need to consider the total number of connections to this Ethernet port: if you run 2 Vijeo Citect redundant servers (that is, 2 OFS instances in parallel), you may encounter a situation when the 2 instances of OFS send their maximum number of parallel requests to the PLC (2 x **Max Pending**). For example, this may happen during redundancy switch over. The PLC cannot service the requests in one cycle so it buffers the requests and services them in several cycles. This happens only in the transient state. During normal and steady state, only 1 server is active and sends requests.

Decreasing the Group Update Rate to Test the Communication Performance

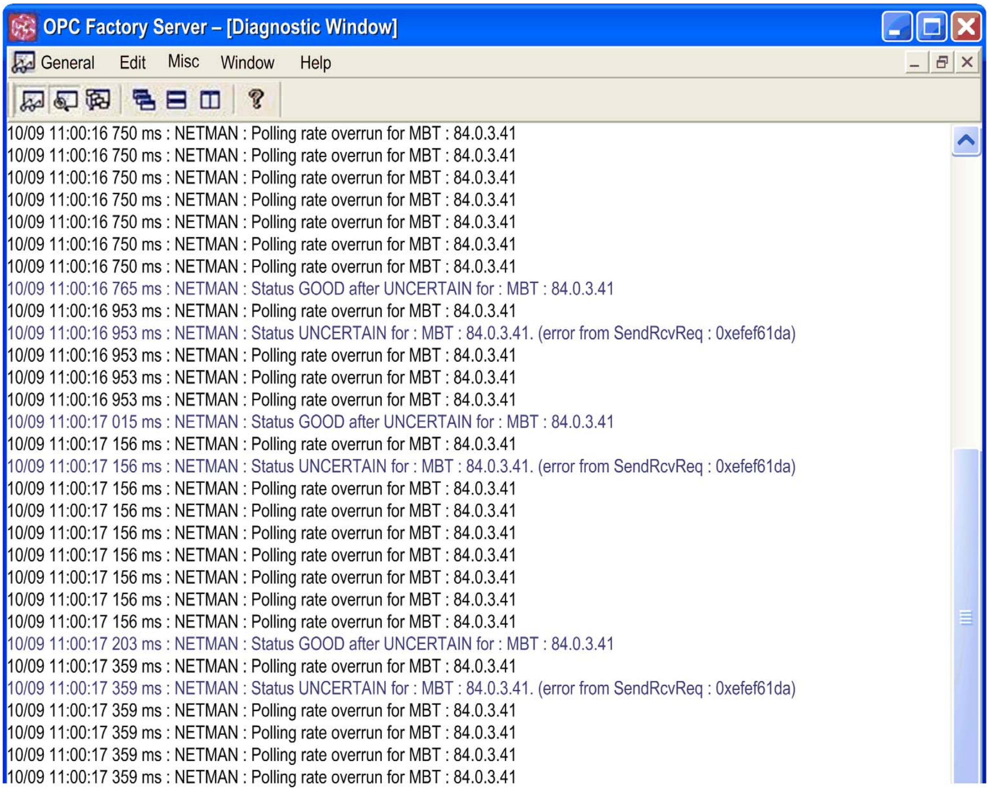
Now that a theoretical value for the group scan time has been set, you can find the minimum OPC client refresh rate (subscription rate). This group update rate was set to 5000 ms earlier. You can try to decrease it to 2 times the group scan time calculated above. This will allow you to check if OFS performs as required and if the above theoretical calculation works in practice. One way to verify whether OFS keeps up with the refresh rate of the client or not is to turn on the Verbose mode in OFS. If some requests are not responded before the next OFS poll cycle, a message appears as **Polling rate overrun for XXX** in the OFS **Diagnostic Window**.

In the above example, the calculated group scan time is 160 ms. You can try to decrease the group refresh rate of the OFS client to 400 ms, and monitor the **Diagnostic Window** in Verbose mode.



```
10/09 10:57:46 250 ms : Debug message box ready.
10/09 10:58:08 765 ms : New update rate (5000 ms, old was 400 ms) for group GRP2
10/09 10:58:16 906 ms : New update rate (400 ms, old was 5000 ms) for group GRP2
```

As shown above, no message appears. This means that OFS can send and receive all 20 requests before the next OFS scan cycle (400 ms). You can now try to decrease the OFS client group refresh rate to 300 ms.



```
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 750 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 765 ms : NETMAN : Status GOOD after UNCERTAIN for : MBT : 84.0.3.41
10/09 11:00:16 953 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 953 ms : NETMAN : Status UNCERTAIN for : MBT : 84.0.3.41. (error from SendRcvReq : 0xefef61da)
10/09 11:00:16 953 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 953 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:16 953 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 015 ms : NETMAN : Status GOOD after UNCERTAIN for : MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Status UNCERTAIN for : MBT : 84.0.3.41. (error from SendRcvReq : 0xefef61da)
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 156 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 203 ms : NETMAN : Status GOOD after UNCERTAIN for : MBT : 84.0.3.41
10/09 11:00:17 359 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 359 ms : NETMAN : Status UNCERTAIN for : MBT : 84.0.3.41. (error from SendRcvReq : 0xefef61da)
10/09 11:00:17 359 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 359 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 359 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
10/09 11:00:17 359 ms : NETMAN : Polling rate overrun for MBT : 84.0.3.41
```

You can see in the above graphic that OFS fails to achieve the desired refresh rate of the client. The value of some of the items therefore becomes uncertain. This value must be avoided being sent to the SCADA system. A general rule is to set the refresh rate of the OPC to 2 times the calculated OFS scan time. Testing is required as shown above to verify the health of the communication channel.

Determining the OPC Client Update Rate

Now that you have finished testing with the OFS test client, follow the same steps with your real OPC client. This helps you determine the realistic update rate that you can specify in your SCADA/OPC client and the performance that you can expect. As mentioned earlier, the **Nb VarMan Req** may be higher in the real SCADA project for the same number of variables to be refreshed because the required item addresses are not contiguous, so OFS needs to send more requests.

The group update rate must be configured in the OPC client directly.

In the Vijeo Citect and the OFS OPC driver, this is done through the following *ini* parameters:

- `Group1 Update Rate`
- `Group2 Update Rate`
- `Group3 Update Rate`

NOTE: OFS does not support group update rate inferior to 300ms.

Glossary

A

Address

Builder name for a PLC variable. For example "%MW1".

Alias

An alias is a shortcut that may be used when a network address for the device is necessary (single replacement string). The use of an alias is also a very practical way to disconnect your OPC application from network addresses of devices that may be modified when necessary.

ASP

Active **S**erver **P**age allows a Web site builder to dynamically create web pages. It supports the code written in compiled languages such as by Visual Basic, C++, C #, etc.

C

CCOTF

Configuration **C**hange **O**n **T**he **F**ly.

Client application

Software using the primitives provided by a server application, via mechanisms (interfaces) implemented by OLE.

CLR

Common **L**anguage **R**untime is part of the .Net framework. It is the program that controls execution of programs written in all supported languages allowing them to understand each other. It also controls the security aspect.

CLS

Common **L**anguage **S**pecification allows the user to optimize and ensure the interoperability of languages by defining all functions that developers may use in numerous languages.

COM

Component **O**bject **M**odel: foundations of the OLE 2.0 standard.

CRA

Communicator **R**emote **A**daptater: drop end communicator.

CRP

Communicator **R**emote **P**rocessor: I/O network head module or bus head communicator.

D

DCOM

Distributed **COM**: COM model distributed over a TCP-IP network.

F

FIP

Factory **I**nstrumentation **P**rotocol.

FTP

File **T**ransfer **P**rotocol is the standard internet protocol that is used for exchanging files between computers and the internet.

G

GAC

Global **A**ssembly **C**ache contains all assemblies necessary for .NET and manages different versions of assemblies.

H

Handle

Single value identifying the object.

HTML

Hyper**T**ext **M**ark-up **L**anguage is the language used to describe Web pages.

HTTP

Hyper**T**ext **T**ransfert **P**rotocol is the protocol used for transferring HTML pages.

I

IDE

Integrated **D**evelopment **E**nvironment is a program that includes a code editor, a compiler, a detected error analyzer and a graphic interface.

IIS

Internet **I**nformation **S**erver is the ftp, Web or HTTP server developed by Microsoft to work under Windows.

Impersonation

Ability to execute a thread with a different security context than that of the thread's owner in a client/server application. When a client contacts a server, the server typically runs with the security context of some service account that has access to every resource that it might possibly need to carry out a request.

J

JRE

Java **R**untime **E**nvironment is a subgroup of the Sun Java development kit that may be embedded in an application. JRE provides minimum conditions (an environment) for running a Java application.

L

LCID

Language **C**ode **ID**entifier.

M

Multi-clients

Several client applications simultaneously access the same server application.

O

OFS

OPC Factory **S**erver: OLE server for exchanging data with the PLC.

OLE

Object **L**inking and **E**mbedding: object for linking and embedding. In particular supplies the OLE Automation interface, a technique which enables a server to display the methods and properties to a client.

OPC

OLE for **P**rocess **C**ontrol.

OPC group

Controls a collection of **OPC** items, that is a list of PLC variables.

OPC item

PLC variable on a PLC and a given communication medium.

OPC server

Controls a collection of OPC groups. Hierarchical root of the OPC model.

P

PLC

Programmable **L**ogic **C**ontroller: programmable controller (industrial).

Primitive

OPC function

R

RCW

Runtime **C**allable **W**rapper: The main function is to group calls between .Net client and the non-managed COM object.

RDE

Read **D**ata **E**ditor: The OFS RDE is used to display and edit the variables of devices from a table based on a Java application or window.

Remote server

The client and server application are located on 2 separate stations, linked by the Microsoft TCP-IP network.

S

Server application

Software presenting the primitives to the client applications, via mechanisms (interfaces) implemented by **OLE**.

SOAP

Simple **O**bject **A**ccess **P**rotocol a Microsoft protocol using HTTP and XML for information exchange.

Socket

Communication channel established between the OFS server and one or more PLCs, on a given communication media. The number of sockets available depends on the communication medium.

SOE

Sequence **O**f **E**vents.

SP

Service **P**ack: operating system corrections and upgrades

Symbol

Identifier attributed by a designer to a control system. For example "PUMP". A symbol cannot start with the prefix ‘%’.

U

UNC

Universal **N**aming **C**onvention.

V

VB

Visual **B**asic: consumer language supporting OLE Automation.

VBA

Visual **B**asic for **A**pplications: Script language using Basic syntax included in the MS-Office Suite.

W

Wintel

Windows/Intel: describes a PC equipped with a 32-bit Windows operating system and an Intel x86 processor.

WSDL

Web **S**ervice **D**escription **L**anguage provides a basic model in XML format for describing Web services.

X

XML

e**X**tensible **M**arkup **L**anguage is an derived extensible meta-language used for structuring data.