

Whitepaper



# Resource Efficiency Testing

Step by Step Instructions

*OPC Foundation*

May 16, 2014

# Contents

Executive Summary .....	3
Intended Audience .....	3
OPC Foundation Membership.....	3
Introduction.....	4
OPC Server Testing.....	4
Multiple data-Sources.....	5
Multiple OPC Clients .....	6
Intermittent Connection Simulation.....	7
OPC Client Testing.....	8
Multiple OPC Servers .....	8
Intermittent connection simulation .....	9
Performance Monitoring.....	10
Windows Performance Monitor.....	10
Linux Performance monitor .....	13
Windows Batch File.....	14

# Executive Summary

All OPC products can be certified (Clients, Servers, or both) for Unified Architecture, OPC Classic, or both.

Most products that fail certification testing do so because of resource leaks or crashes found in the 36-hour resource efficiency testing. This document provides step-by-step instructions for you to replicate the resource efficiency testing conducted in the OPC Foundation Certification Test Lab to help you avoid this problem.

This paper covers the following:

- Configuring your OPC Client/Server
- Configuring products to connect to your Client/Server
- Configuring the environment for simulating network communications outages
- Obtaining and analyzing performance metrics

Vendors need to perform this test on their Client/Server products before submitting to the test lab.

## Intended Audience

This document contains technical information and is intended for software developers and QA personnel.

## OPC Foundation Membership

This document is primarily intended for members of the OPC Foundation.

---

Copyright © 2014 OPC Foundation. All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any language, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission from the OPC Foundation.

All copyright, confidential information, patents, design rights, and all other intellectual property rights of whatsoever nature contained herein are and shall remain the sole and exclusive property of the OPC Foundation.

The information furnished herein is believed to be accurate and reliable. However, no responsibility is assumed by the OPC Foundation for its use or for any infringements of patents or other rights of third parties resulting from its use. All other trademarks are the property of their respective owners.

# Introduction

A 36-hour resource-efficiency test will combine stress-testing with robustness and recovery. The goal is to ensure products are reliable, able to automatically recover from communications loss, and are efficient with system resources (no memory-leaks etc.)

The general testing principle is the same for OPC Clients and Servers. However, some subtle differences require us to treat them differently. Please review the applicable sub-section as applicable to your product.

## OPC Server Testing

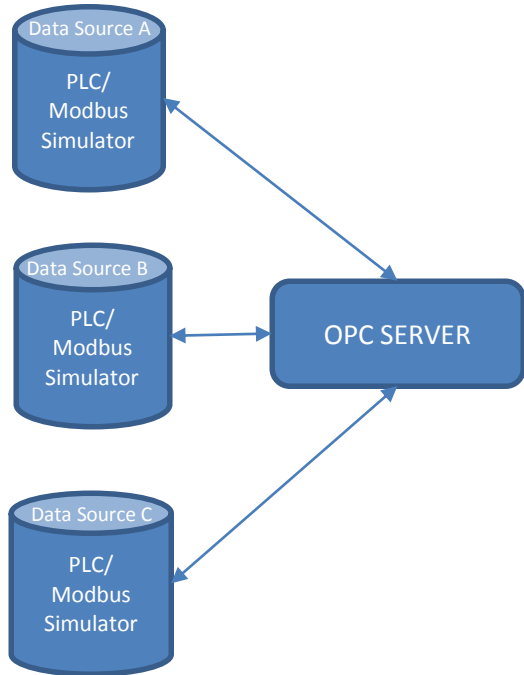
You will need the following:

- Multiple data-sources (PLCs, Modbus Simulators etc. if supported)
- Multiple OPC Clients (5, by minimum)
- Windows Performance Monitor (See *section Performance Monitor on page 9*)
- Intermittent network connection (simulation)

## Multiple data-Sources

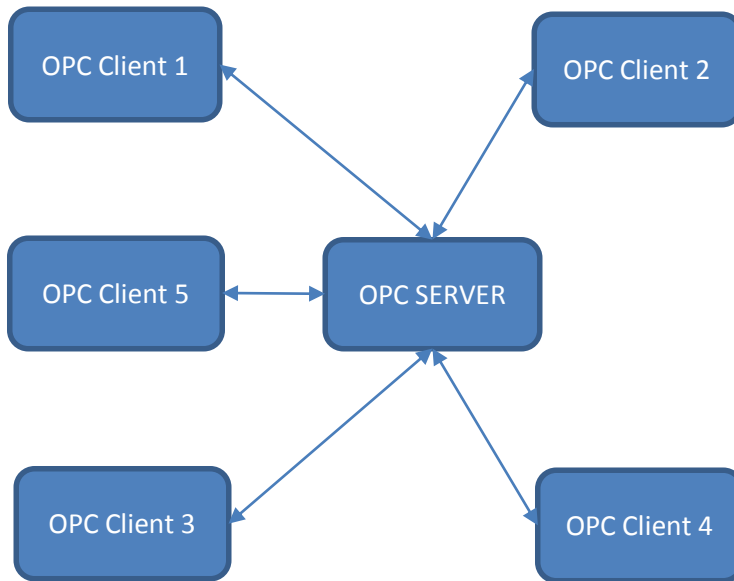
*Note: you can skip this section if your server does not support Device connections*

- The Test lab uses “PeakHMI TCP Slave” / [MODBUS Ethernet PLC simulator](#).
- Multiple data-sources are needed (at least 2 remote data sources) as shown here:



## Multiple OPC Clients

- A minimum of 5 OPC clients' are needed:



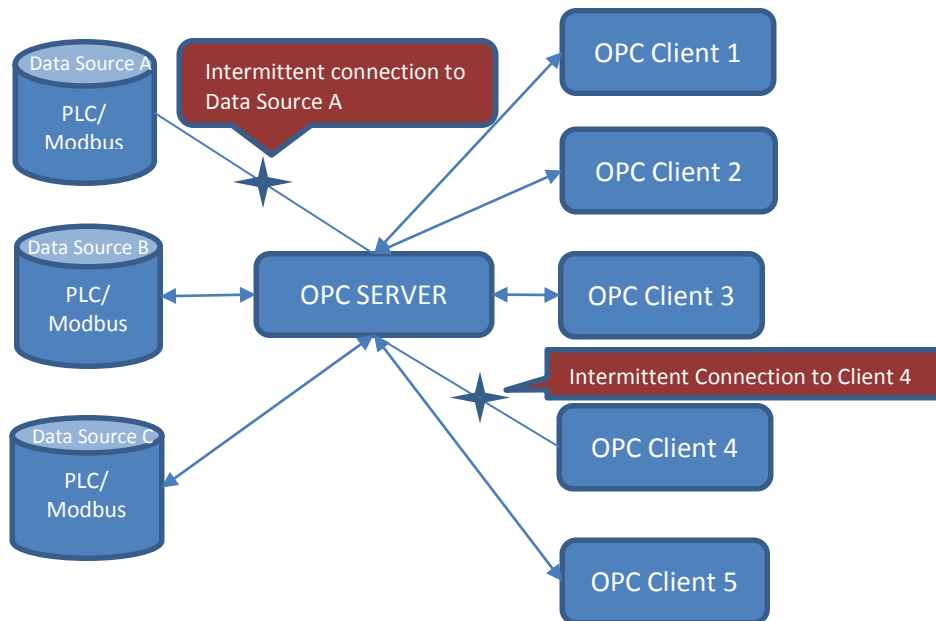
- Each client must consume a minimum of 1000 items/nodes from each Server covering all devices.
- If the server does not support device connections, configure the client with all the available items (internal/simulation tags) in the server
- A list of typical Classic/UA Clients used in the test lab is shown here:

<b>Classic</b>	Iconics Genesis64 <i>V 10.81+FH2</i>	Indusoft- WebStudio <i>V 7.1</i>	Softing Demo Client <i>V 4.30.1 Build 2692</i>	OPCF DA Sample Client <i>V 334</i>	Kepware-OPC Quick Client <i>V 5.11.250.0</i>
<b>UA</b>	Iconics Genesis64 <i>V 10.81+FH2</i>	Indusoft- WebStudio <i>V 7.1</i>	HB-Soft Comet UA Browser <i>V 1.0.3</i>	OPCF UA Sample Client <i>V 1.1.334.0</i>	Unified Automation- UA Expert <i>V 1.2.2</i>

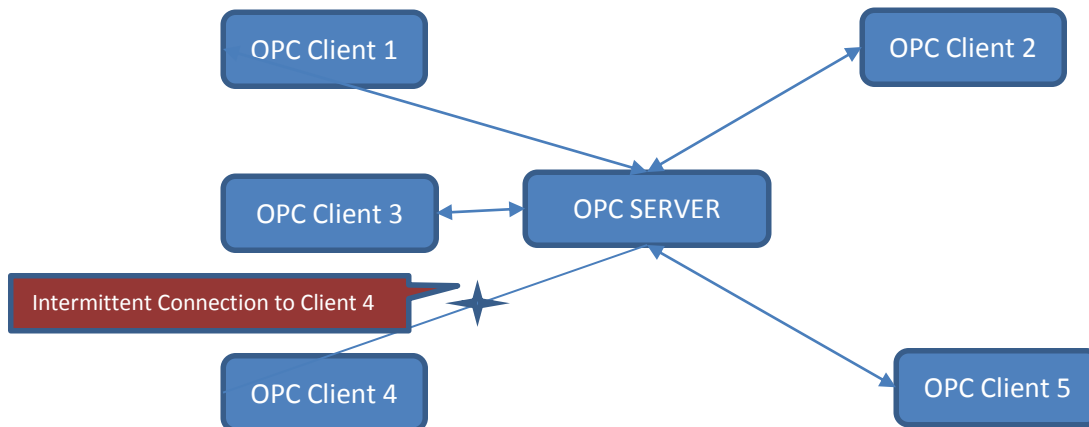
*Please contact the vendors directly to obtain the demo version of their software.*

## Intermittent Connection Simulation

- Configure the performance monitor tool to record the data as mentioned in *Performance Monitor section on page 9*.
- Let the system run for an hour without any interruption to get a base-line of resource consumption.
- Run the batch file (*see section Batch File on page 13*) to start simulating intermittent network connectivity.
- There are two different sets of configuration for the intermittent connection. Please use the configuration applicable for your application:
  - **Configuration 1:** When the server supports device connections:



- **Configuration 2:** When the server is not connected to an external data source:



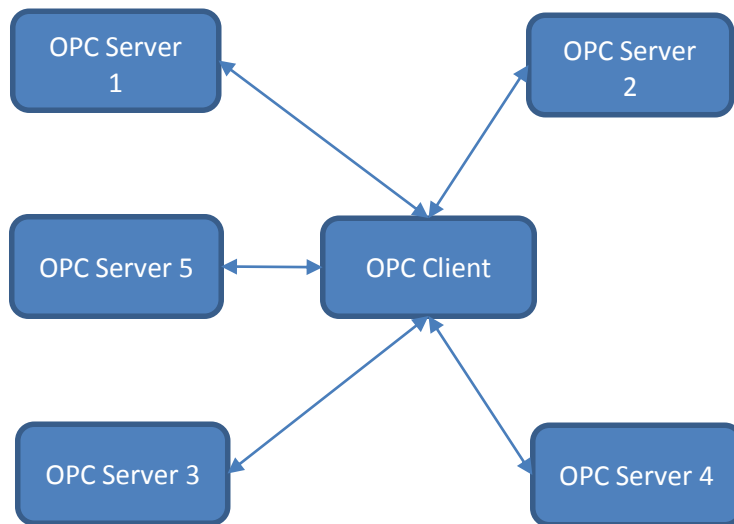
# OPC Client Testing

You need to have the following in order to perform the resource efficiency test on a Client.

- Multiple OPC Servers (5, by minimum)
- Windows Performance Monitor (See *section Performance Monitor on page 9*)
- Intermittent connection (simulation)

## Multiple OPC Servers

- A minimum of 5 Servers need to be used for the efficiency test:



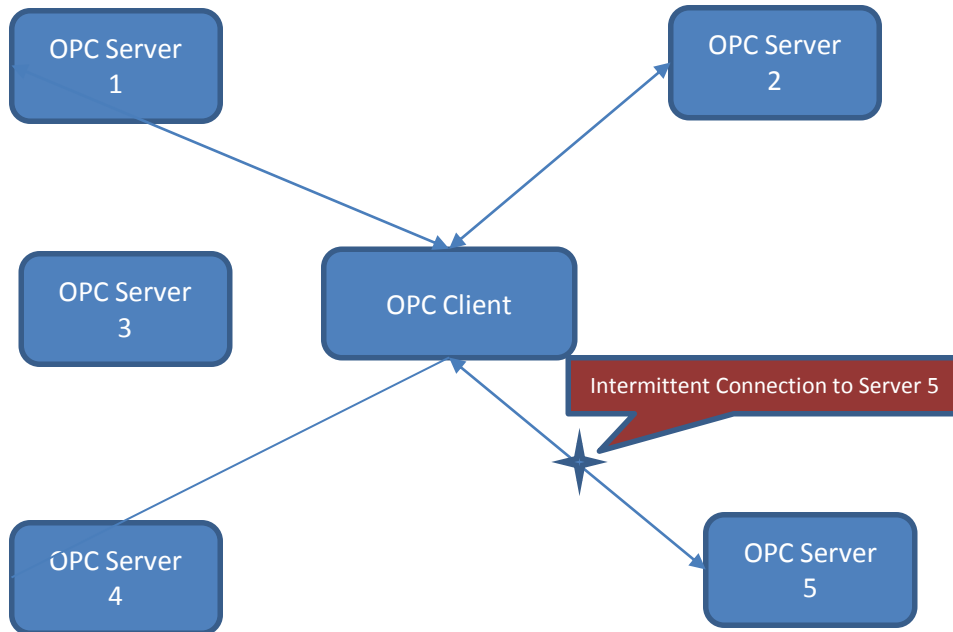
- Connect the client to each server as shown and configure 1000 items per OPC server connection.
- A list of typical Classic/ UA servers used in the test lab are shown in the below table

<b>Classic</b>	Kepware KepServerEX V 5.11	Matrikon UCS V 2.1.0.1602	Softing-OPC Toolbox Demo Server V 4.30.1 Build 2692	OPCF Sample Server V 1.0.01	Cyberlogics- MBX OPC Server V 7.1
<b>UA</b>	Kepware KepServerEX V 5.11	Matrikon UCS V 2.1.0.1602	Softing – OPC UA C++ Test Server V 5.20.0.8536	OPCF UA Reference Server V 1.1.334.0	UnifiedAutomation- UA Demo Server V 1.3.2.200



## Intermittent connection simulation

- Configure the Client to connect to multiple Servers, as shown here:



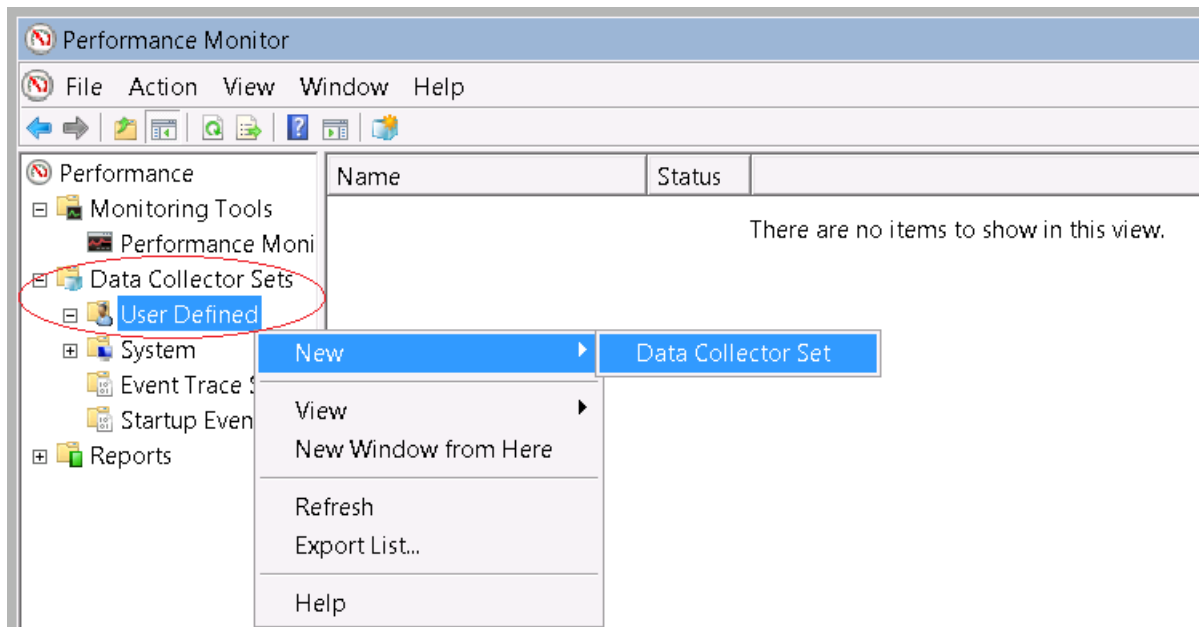
- Configure the performance monitor tool to record the data as mentioned in *Performance Monitor section on page 9*.
- Let the system run for an hour without any interruption to get a baseline of resource consumption.
- Run the batch file (*See section Batch File on page 13*) to enable the intermittent connection simulation

# Performance Monitoring

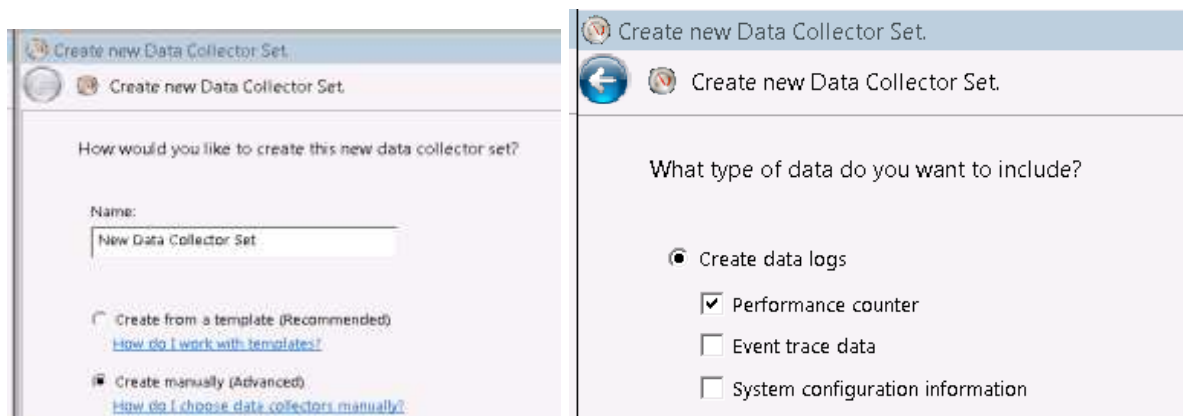
## Windows Performance Monitor

Windows performance monitor is used to record performance data and is then converted to charts for reporting.

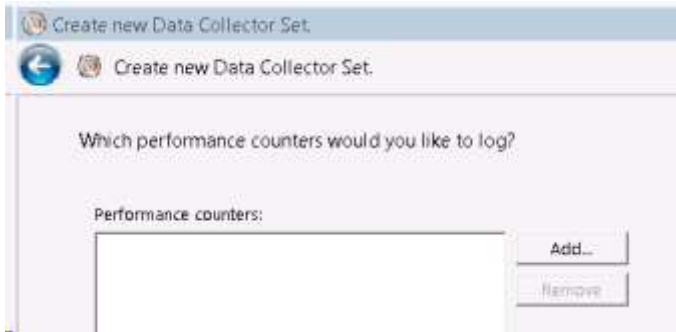
1. Open Performance monitor by typing “Perfmon.msc” in windows Start menu search.
2. Create a new collector set in the “Data Collector Sets/ User Defined” section of the performance monitor:



3. Select “Create Manually” option on the screen and select Performance counter under “Create data logs” option on the next screen:



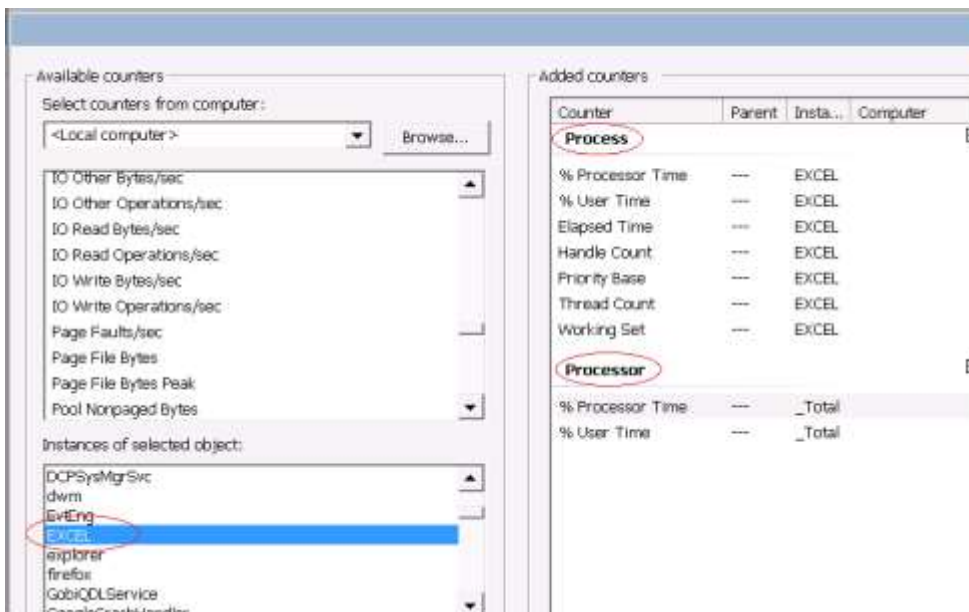
- Select the performance counters in the next screen. Click “Add” to include the counters that needs to be monitored:



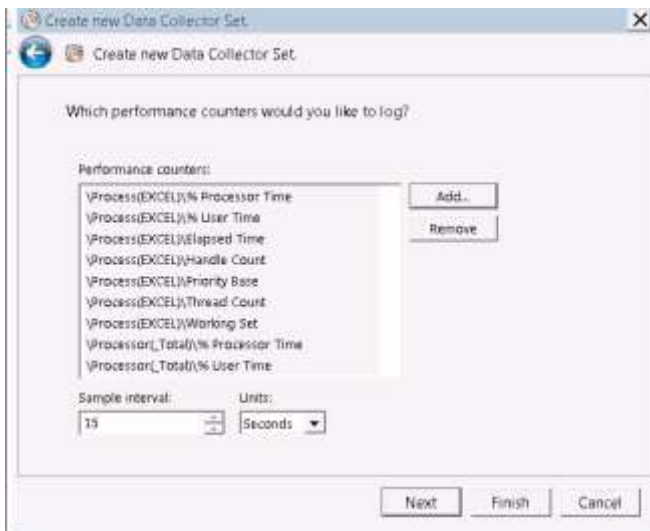
- Here is the list of counters needed for the test:

	Counters	Instances
<b>Processor</b>	<ul style="list-style-type: none"> <li><input type="radio"/> % Processor Time</li> <li><input type="radio"/> % User Time</li> </ul>	_Total
<b>Process</b>	<ul style="list-style-type: none"> <li><input type="radio"/> % Processor Time</li> <li><input type="radio"/> % User Time</li> <li><input type="radio"/> Elapsed Time</li> <li><input type="radio"/> Handle Count</li> <li><input type="radio"/> Priority Base</li> <li><input type="radio"/> Thread Count</li> <li><input type="radio"/> Working Set</li> </ul>	All related Instances of your Opc Server/ OPC Client

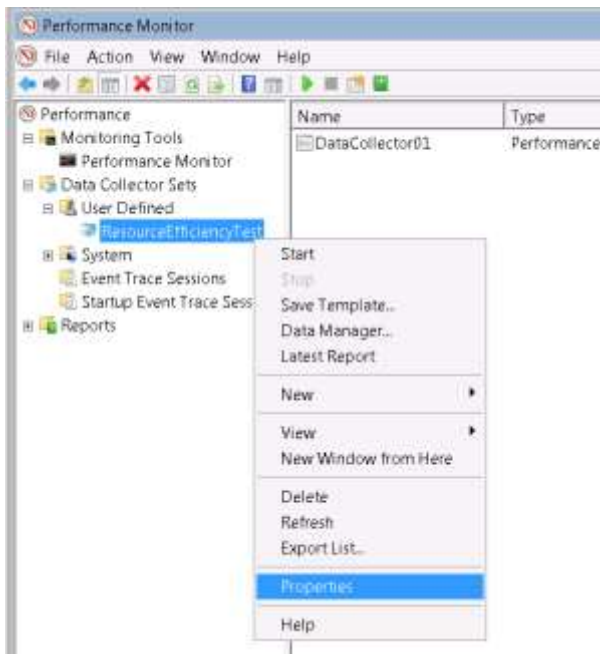
- Add the counters listed above, as shown here:



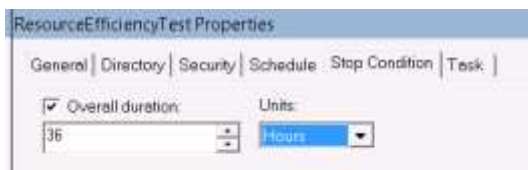
7. Set the Sampling Interval, e.g. 15-seconds, and then follow the steps to create the collector set:



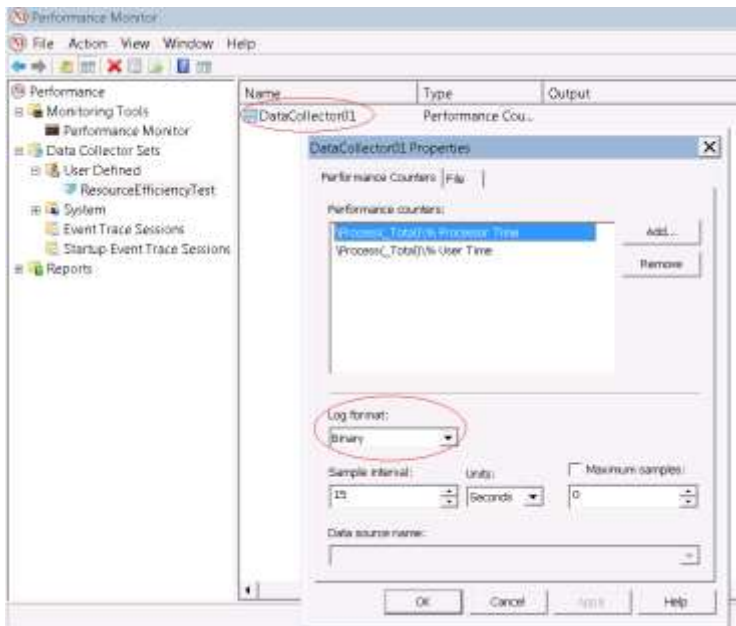
8. Right click on the Data collector set and choose properties to set the Stop Condition:



9. To record the data for 36 hours specify the “overall duration” as 36 hours in the “stop Condition” and click “ok”:



10. Right click on the DataCollector01 to see the DataCollector01 properties.



11. Select the log format and click "ok".

12. Start the collector once the OPC Server/ OPC Client is ready to start the 36-hour test.

## Linux Performance monitor

- Linux provides the **top** command to monitor the performance during the efficiency test.
- Below is the screenshot of how the **top** command looks in the terminal:

```
top - 00:23:58 up 8 min, 2 users, load average: 0.36, 0.55, 0.28
Tasks: 118 total, 3 running, 115 sleeping, 0 stopped, 0 zombie
Cpu(s): 6.0%us, 2.0%sy, 0.0%ni, 92.0%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1026436k total, 478668k used, 547768k free, 15692k buffers
Swap: 1646620k total, 0k used, 1646620k free, 216820k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3287	root	20	0	143m	66m	19m	S	0.0	6.7	0:21.38	firefox
3752	root	30	10	85676	61m	25m	S	0.0	6.1	0:04.67	update-manager
2463	root	20	0	123m	30m	9888	R	3.0	3.0	0:32.80	Xorg
3282	root	20	0	40420	23m	13m	S	1.7	2.3	0:06.65	gnome-panel
3283	root	20	0	54984	18m	13m	S	0.0	1.8	0:02.32	nautilus
3413	root	20	0	43748	13m	10m	S	0.0	1.4	0:00.58	mixer_applet2
3293	root	20	0	27748	13m	8044	S	0.0	1.4	0:00.30	python
3295	root	20	0	27320	13m	10m	S	0.0	1.4	0:00.32	update-notifie
3677	root	20	0	32800	13m	9216	S	1.0	1.3	0:00.90	gnome-terminal

- Use the following command in the terminal to view the output:  
`./top_pname.sh <OPC Server/ OPC Client>`
- Use the following command to redirect the output to a csv file:  
`./top_pname.sh <OPC Server/ OPC Client> &> <Filename.csv>`
- Use Ctrl+c to terminate the process.

## Linux TOP Shell Script

The following script will remove unwanted information from the TOP output:

```
#!/bin/bash

if [ $# = 0 ]
then
    echo "arg error : ?"
    echo "top_pname.sh <exename>"
    exit
else
    echo ""
fi

h1="Time,Mem total,Mem used,Mem free,Mem buffers,PID,USER,PR,NI,VIRT,RES,SHR,S,%CPU,%MEM,TIME+,COMMAND"
pname=$1
echo -e "$h1"
print_vals(){
    l1=`/usr/bin/top -bnlp $(pidof $pname)|sed -n -e 1p|awk '{print $3,"",""}'`
    l2=`/usr/bin/top -bnlp $(pidof $pname)|sed -n -e 4p|awk '{print $2,"","$4","","$6","","$8",""}'`
    l3=`/usr/bin/top -bnlp $(pidof $pname)|sed -n -e 8p|awk '{print $1,"","$2","","$3","","$4","","$5","","$6","","$7","","$8","","$9","","$10","","$11","","$12}'`
    echo -e "$l1$l2$l3"
}

while true
do print_vals
  sleep 4
done
```

## Windows Batch File

Test lab uses a batch file to enable/disable the “Local Area Connection” network adapter in a windows environment. Copy the below text into a “.bat” file and run it using Administrator/elevated permission to enable the intermittent connection.

```
@echo off
SET DTIME = 120
CLS
ECHO Delay time is: %DTIME%
ECHO Network Connection Enabler/Disabler started at: %date% %time%
:start
ECHO Disabling network adapter at %time%
netsh interface set interface "Local Area Connection" DISABLED
timeout /t %DTIME%
ECHO Enabling network adapter at %date% %time%
netsh interface set interface "Local Area Connection" ENABLED
choice /c:yn /m "Repeat network adapter recycle (y/n)?" /t %DTIME% /d y
ECHO %ERRORLEVEL%
IF ERRORLEVEL 2 GOTO end
IF ERRORLEVEL 1 GOTO start
:end
ECHO.
ECHO Network adapter recycle complete at %date% %time%
```

You can change the time period of the intermittent connection by varying DTIME in the above code.



**Contact Information:**

OPC Foundation

16101 N. 82<sup>nd</sup> Street, Suite 3B

Scottsdale, AZ 85260

USA

Tel: 480-483-6644

Fax: 480-483-1830

Email: [compliance@opcfoundation.org](mailto:compliance@opcfoundation.org)



Page Intentionally Left Blank





## Revision History

Version	Date	Author	Change
1	May 16, 2014	DT	Initial version.