## **OPC UA: How many variables is the limit?**

The title is a frequently asked question that can be answered quite easily: "There is no fixed limit." Nevertheless, there are various possibilities to minimize the performance demand of the OPCUA server and thus to lighten the load on the controller. The influence of hardware is ignored in this article.

All values refer to the online mode. Load peaks during login or browsing of the data points via the client are not examined.

- A Raspberry Pi 3 serves as controller and the Unified Automation UaExpert was used as client.
- A POU with 1000 variables of type UDINT as created as a test project.
- A value change can be switched on or off at the variables:



<u>/</u>]

This POU can then be copied as often as required, working with 10,000 "prepared" variables. The copied POUs are called using the code of  $PLC_PRG$ .



Simply providing the data points has no negative influence on the CPU load because the data is not used. However, the compiler process takes longer. In the test project, the CPU has a load of ~8%, whereby the monitoring of the CODESYS IDE is continuously active.

🛃 pi@ThK_raspberry: ~	
top - 08:58:29 up	2:58, 1 user, load average: 1.12, 1.08, 1.05
Tasks: 112 total,	1 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.6 us,	2.2 sy, 0.0 ni, 97.1 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
KiB Mem: 882500	total, 153200 used, 729300 free, 13528 buffers
KiB Swap: 0	total, 0 used, 0 free. 79600 cached Mem
PID USER PR	NI VIRT RES SHR S SCPU SMEM TIME+ COMMAND
713 root 20	0 29876 23988 2228 S 7.9 2.7 19:57.22 codesyscontrol.
222 root 20	0 0 0 0 0 <del>6.6</del> 0.0 12:27.93 rcio_worker
197 root 20	0 0 0 0 5 1.3 0.0 2:30.92 spi1
979 pi 20	0 5112 2540 2164 R 0.7 0.3 1:09.01 top

As soon as the client is connected, the load increases to ~9%, which is the base load.

W Unified Automation UaExpert - The OPC Unified Architecture Client - PerformanceTest_1000*													
<u>File View Server D</u> ocument <u>S</u> ettings <u>H</u> elp													
🗋 🖸 🎜 🛛 🙆 💿 💠 🛏 🔌 🙎		R	<b>2</b>										
Project 🗗 🗶	Data A	ccess View											
🔺 📁 Project	#			Se	rver								
4 📁 Servers													
📎 OPCUAServer@ThK_raspberry - None - None (uat	🗞 OPCUAServer@ThK_raspberry - None - None (uat												
4 🗊 Documents													
Data Access View													
top - 08:59:51 up	2:59, 1 user, load average: 1.09, 1.08, 1.05												
Tasks: 112 total,	1 running, 111 sleeping, 0 stopped, 0 zombie												
%Cpu(s): 0.8 us,	2.0 s	sy, 0.0	ni, 97.	2 id,	0.0 wa, 0.0	) hi, 0.0 si, 0.0 st							
KiB Mem: 882500	total	153	200 used	1, 729	300 free,	13560 buffers							
KiB Swap: 0	total		0 used	i,	0 free.	79600 cached Mem							
Address Space PID USER PR	NI	VIRT	RES	SHR S	SCPU SMEM	TIME+ COMMAND							
No Highlight 713 root 20		29876	23988	2228 S	9.6 2.7	20:03.95 codesyscontrol.							
222 root 20				0 D	6.9 0.0	12:33.54 rcio_worker							
197 root 20	0	0	0	0 S	1.3 0.0	2:32.07 spi1							
979 pi 20	0	5112	2540	2164 R	0.7 0.3	1:09.56 top							

## Value changes

When the first 1000 data points are activated by the client, the load increases to ~17%.

Data	Access View																	0
#			Ser	ver						Node I	id			Display Nar	me	Value	Datatype	Gource Timestam 📩
1	OPCUASen	er@ThK rasp	berry - No	ne - None	(uatcp-uasc	-uabinary)	NS4IStringIlva	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC PRG.ud	di000	udi000	0		UInt32	11:04:19.769 =
2	OPCUASen	er@ThK_rasp	berry - No	ne - None	uatep-uase	-uabinary)	NS4IStringIlva	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC PRG.uc	di001	udi001	0		UInt32	11:04:19.769
3	OPCUASer	er@ThK_rasp	berry - No	ne - None	uatcp-uasc	-uabinary)	NS4 String  vai	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC PRG.ud	di002	udi002	0		UInt32	11:04:19.769
4	OPCUASen	er@ThK rasp	berry - No	ne - None	(uatcp-uasc	-uabinary)	NS4 String  var	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC PRG.uc	di003	udi003	0		UInt32	11:04:19.769
5	OPCUASen	er@ThK_rasp	berry - No	ne - None	(uatcp-uasc	-uabinary)	NS4 String  var	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC_PRG.uc	di004	udi004	0		UInt32	11:04:19.769
6	OPCUASen	er@ThK_rasp	berry - No	ne - None	(uatcp-uasc	-uabinary)	NS4 String  vai	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC_PRG.uc	di005	udi005	0		UInt32	11:04:19.769
7	OPCUASen	er@ThK_rasp	berry - No	ne - None	(uatcp-uasc	-uabinary)	NS4 String  vai	CODESYS	Control for I	Raspberry Pi S	SL.Applicatio	n.PLC_PRG.ud	di006	udi006	0		UInt32	11:04:19.769
8	CODCUMC	ATHY AND	learner Mar	na Mana	location come		MCAICAsin allow	CODECVE	Controlferel	D b D: (	N Annibustin	- 01 C 00C	1:007		-		-Hint32	11:04:19.769
9	pi@ThK	raspberry: ~		-	-	-	the second secon		-	fragment (* 1	1.000		_	-			📕 ht32	11:04:19.769
10		-	-	-					-			-					ht32	11:04:19.769
11	top - 09	:04:38 up	3:04,	1 use	r, load	average	: 1.24, 1.	.3, 1.07	7								▲ ht32	11:04:19.769
12	Tasks: 1	12 total,	1 ru	nning,	111 slee	ping,	0 stopped,	0 zon	nbie								nt32	11:04:19.769
13	<pre>%Cpu(s):</pre>	2.7 us.	3.1 s	v. 0.0	ni, 94.	2 id. 0	.0 wa. 0.0	) hi. (	0.0 si.	0.0 st							nt32	11:04:19.769
14	KiB Mem.	88250	0 total	153	480 11966	7290	20 free	13588	huffers								ht32	11:04:19.769
15	Wild Com		01	, 200	0	, ,200	0 5	20000	Durrer of	(							nt32	11:04:19.769
10	KID Swap		U LOLAI		U used		U free.	/9600	caened 1								nt32	11:04:19.769
1/																	1132	11:04:19.769
18	PID US	ER F	R NI	VIRT	RES	SHR S	SCPU SMEM	TIM	IE+ COMM	AND							1132	11:04:19.769
19	713 ro	ot 2	0 0	29876	23988	2228 S	16.5 2.7	20:38.	.03 codes	syscontro	1.						1132	11:04:19:709
20	222 ro	ot 2				0 0	7300	12.53	21 rcio	worker							102	11:04:19:709
22	107 20	a+ 2	0 0			0 9	1 2 0 0	2.26	0.0 april								+22	11.04.15.709
22	197 10	2	0 0		05.40	0.5	1.5 0.0	2:30.	.05 Spil								n+32	11:04:15:709
123	979 pi	2	0 0	5112	2540	2164 R	0.7 0.3	1:11.	.50 top								ICJ2	11.04.15.705

When we start the counter, we have a constant value change of the variable. This increases the load to  ${\sim}23\%.$ 

Data	Access View																		0
#			Ser	ver							Node Id			Display Name	Va	lue	Datatype	Jource Timestar	n 🔺
1	OPCUAServer@	ThK raspbe	erry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4IStri	nallvarl	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PR0	G.udi000	udi000	1074		UInt32	11:07:35.100	=
2	OPCUAServer@	ThK raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	ngllvarl	CODESYS Cov	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PRO	G.udi001	udi001	1074		UInt32	11:07:35.100	-
3	OPCUAServer@	ThK raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	ngllvarl	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PRO	G.udi002	udi002	1074		UInt32	11:07:35.100	
4	OPCUAServer@	ThK raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	ngllvarl	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PRO	G.udi003	udi003	1074		UInt32	11:07:35.100	
5	OPCUAServer@	ThK_raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	nglivari	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PRO	G.udi004	udi004	1074		UInt32	11:07:35.100	
6	OPCUAServer@	ThK_raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	nglivari	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PRO	G.udi005	udi005	1074		UInt32	11:07:35.100	
7	OPCUAServer@	ThK_raspbe	rry - No	ne - None	(uatcp-uaso	c-uabinary)	NS4 Stri	ng  var	CODESYS Cor	ntrol for Rasp	berry Pi SL.Appl	cation.PLC_PR0	G.udi006	udi006	1074		UInt32	11:07:35.100	
8 1	ODCUAC	TI-V	Man Nia	M	(		NICALCA-	- all - a all	CODECVC C-	D	Is a new Di Cl. A se se l	DLC DDC	C		1074		Hnt32	11:07:35.100	
9	pi@ThK_rasp	oberry: ~		-	-							and the second		-			1 nt32	11:07:35.100	
10					-		1000	-				-		_	-	_	nt32	11:07:35.100	
11	top - 09:07	:33 up	3:07,	1 use	r, load	d average	: 1.30	, 1.1	6, 1.09							^	nt32	11:07:35.100	
12	Tasks: 111	total,	1 ru	unning,	110 slee	eping,	0 stop	0 stopped, 0 zombie								nt32	11:07:35.100		
13	%Cpu(s): 3	.8 us.	2.1 s	v. 0.0	ni, 93.	.5 id. 0	.0 wa.	0.0	hi, 0.7	si, 0.	0 st						nt32	11:07:35.100	
14	KiB Mem.	882500	total	159	348 1190		52 fre	•	13636 bi	ffers							nt32	11:07:35.100	
15	WiD Summe	002000	+-+-1	, 100	0 10 0000	~ /2.51	0 6	_,	70604 -	abad Mam							nt32	11:07:35.100	
10	KID Swap:	0	LOLAI		U used		0 Ire	е.	/9604 Ca	iched Mem							nt32	11:07:35.100	
1/																	nt32	11:07:35.100	
18	PID USER	PR	NI	VIRT	RES	SHR S	%CPU	%MEM	TIME+	<ul> <li>COMMAND</li> </ul>							1132	11:07:35.100	
19	713 root	20	0	29876	23988	2228 S	22.8	2.7	21:08.43	codesys	control.						1132	11:07:35.100	_
20	222 root					0 0	7.3	0.0	13:05.58	rcio wo	rker						102	11:07:35.100	
21	107 root	20				0 5		0.0	2.20 54								+22	11:07:55.100	
22	197 root	20	0	5440	05.40	0.00	1.7	0.0	2.30.34	Spil							nt32	11:07:35:100	
24	9/9 pi	20	0	5112	2540	2164 R	0.7	0.3	1:12.69	COD							nt32	11:07:35.100	

The measurement results with an increasing number of data points are summarized in the following table:

Number of data points	Subscribed data points without value change	Subscribed data points with value change
0	~10% (base load) of the project	~10% (base load) of the project
1000	~17%	~23%
2000	~25%	~35%
3000	~30%	~45%
4000	~36%	~55%
5000	~41%	~67%

A The first conclusion can be drawn as follows:

• The CPU load, and therefore also the number of variables, depends on the number of value changes.

The next point, the sampling rate, can also be derived from this.

## Sampling rate

The above measurement results are recorded with the default settings of the OPCUA client with an refresh rate of 500 milliseconds. There are many values and parameters for which a slower refresh rate at the client has practically no influence. As an example, a room temperature or preset/setpoints, such as the parameters of a PID controller, should be mentioned here.

Starting from the worst case in the above table, more and more data is now set to a lower sampling rate, while the others remain unchanged:

Sampling rate / Number of data points	0	1000	2000	3000	4000	5000
1000	~67%	~65%	~64%	~62%	~60%	~59%
2000	~67%	~64%	~62%	~59%	~58%	~56%
5000	~67%	~64%	~61%	~59%	~56%	~54%

A mixed operation of 1000 variables each with a sampling rate of 500, 1000, 2000, 3000, and 4000 milliseconds resulted in a CPU load of ~59%.

The second conclusion can be drawn as follows:

The CPU load can also be reduced by dividing the variables into groups with different refresh rates.

The variables of a project can be combined not only in time-based groups via the OPCUA client, but also by data type on the controller.

For demonstration purposes, the program is extended a little bit:

UA_Performance.project* - CODESYS		
<u>File Edit View Project Build Online Debug Tools</u>	Window	Help
🎦 🚅 🔚 😂 🗠 🗠 🐰 ங 🛍 🗙 🛤 🎲 🕼	🌢 🚰   a•b	🔺 🐪 🎋   🛍   🛅 + 👔   🕮   🧐 🚯 🕞 🕞 👋   💷 🤒
Devices 🗸 🗸 🗙	PLC	_PRG 🔐 Device 📄 DataArray 🗙 📲 Symbol Configuration
UA_Performance	1	PROGRAM DataArray
Device (CODESYS Control for Raspberry Pi SL)	⊟ 2	VAR CONSTANT
⊨	3	<pre>c_iMaxArray : INT := 1000;</pre>
Application	4	END_VAR
Library Manager	<b>⊟</b> 5	VAR
DataArray (PRG)	6	udiCnt : UDINT;
	7	audi1 : ARRAY [0c_iMaxArray] OF UDINT;
	8	iIndex : INT;
	9	xChange : BOOL;
PLC_PRG_2 (PRG)	10	END_VAR
PLC_PRG_3 (PRG)		
PLC_PRG_4 (PRG)	1	<pre>udiCnt := udiCnt + 1;</pre>
PLC_PRG_5 (PRG)	<b>⊟</b> 2	IF xChange THEN
PLC_PRG_6 (PRG)	<b>⊟</b> 3	FOR iIndex := 0 TO c_iMaxArray DO
PLC_PRG_7 (PRG)	4	<pre>audi1[iIndex] := udiCnt;</pre>
PLC PRG 8 (PRG)	5	END_FOR
	L 6	END_IF
Symbol Configuration		

This time as well, the values are recorded with and without the value being changed. Again, the sampling rate is at 500 milliseconds.



Number of data points	Without value change	With value change
1000	~10%	~11%
2000	~10%	~12%
3000	~11%	~13%
4000	~11%	~14%
5000	~12%	~14%

 $\odot$ 

Even if combining data points in different groups/arrays means more work in the actual project, this investment should be made for larger plants.

()	Please note that all tests were pe Of course, every additional client Here the first measurement (100	erforme also ir D indivi	ed with only one clier acreases the CPU lose dual variables with a	nt connected. ad. I value change) with two	connected OPCUA clients	s:		
	Wunified Automation UaExpert - The OPC Unified Arch	itecture Clie	ent - PerformanceTest_1000	🔚 Unified Automation UaExpert - The C	DPC Unified Architecture Client - Performance	lest_10	00	
	File View Server Document Settings Help			File View Server Document Se	ttings Help			
	🗋 💋 🕞 🙆 🕘 🔶 🗙	2	8 🛛 🧇	🗋 🖉 🖯 🖉 🍳 🌵	= 🔉 🗙 🔦 🔔 🖹 🕱 <			
	Project	Ð×	Data Access View	Project	8 ×	Data	Access View	
	Project     Servers     OPCUAServer@ThK_raspberry - None - N     Documents     Data Access View	lone (uatcp-	OPCUAServer@ThK_raspb     OPCUAServer@ThK_raspb	Project     Servers     OPCUAServer@ThK_rasp     Documents     Data Access View	iberry - None - None (uatcp-uasc-uabinary)	# 1 2 3 4 5 6 7 8	Server OPCUAServer@ OPCUAServer@ OPCUAServer@ OPCUAServer@ OPCUAServer@ OPCUAServer@ OPCUAServer@	NS4 String var CODESYS Control for NS4 String var CODESYS control for
		pi@T	hK_raspberry: ~				and the second s	
	۲. III. III. III. III. III. III. III. I	top -	12:37:18 up 6:37, 1 us	er, load average: 1.33, 1.2	9, 1.20			
	Address Space	Tasks: %Cpu(s	112 total, 1 running, ): 3.7 us. 1.9 sv. 0.	111 sleeping, 0 stopped, 0 ni. 94.1 id. 0.0 wa. 0.0	0 zombie hi. 0.3 si. 0.0 st			
	😏 No Highlight	KiB Me	m: 882500 total, 15	6100 used, 726400 free,	15756 buffers			
	Content Conten	KiB Sw	ap: 0 total,	0 used, 0 free.	79660 cached Mem			
	4 👶 DeviceSet	713	root 20 0 30232	24540 2228 5 23.8 2.8	106:31.13 codesvscontrol.			
	4 💑 CODESYS Control for Raspberry Pi SL	222	root 20 0 0	0 0 D 6.6 0.0	27:30.67 rcio_worker			
	4 💑 Resources	197	root 20 0 0	0 0 5 1.7 0.0	5:31.53 spi1			
	Application	979	pi 20 0 5248	2540 2164 R 1.0 0.3	2:38.48 top			