Monitorovací a vizualizační systém ProCop 3.7

Tipy a triky Vývojářská příručka



© ALFA Mikrosystémy, s.r.o. Ostrava 2022

Tipy a triky

Vývojářská příručka

ALFA Mikrosystémy, s.r.o.

Monitorovací a vizualizační systém ProCop 3.7 je specializovaný software pro monitorování technologických procesů, uživatelské řízení technologií, archivaci historických trendů technologických veličin a alarmních stavů.

Zahrnuje zkušenosti z předchozích verzí systému, které mají počátky již v roce 1993, kdy byl poprvé nasazen první předchůdce stávajícího monitorovacího systému. Dlouholetou praxí v oboru monitorování technologíí, zejména pak v tepelném hospodářství, se podařilo dle zkušeností a požadavků zákazníků a samotných dispečerů vyvinout produkt, který se Vám v tuto chvíli dostává do rukou.

Copyright © 2022 ALFA Mikrosystémy, s.r.o. Ostrava

Microsoft, Windows, Windows 7, Windows Vista, Windows XP, Windows 2000 a Internet Explorer jsou registrované obchodní známky Microsoft Corporation, Intel je registrovaná obchodní známka, Pentium je obchodní známka Intel Corporation. ProCop je registrovaná obchodní známka firmy ALFA Mikrosystémy, s.r.o.

Vytištěno: leden 2022

Obsah

5

Obsah

1 N	Aoxa NPort v UDP režimu	7
1.1	Virtuální port vs. UDP režim	7
1.2	Nastavení UDP režimu	8
	Připojení k převodníku	8
	Konfigurace IP adresy	9
	Konfigurace UDP rezimu Nastavaní sáriováho portu	. 11
	Wastaveni serioveno portu	13
1.3	Úpravy projektu pro N-Port	13
1.4	Problémy s komunikací	15
2 A	utomatické vlastnosti potrubí	17
3 A	tributy knihovních bloků	<u>19</u>
3.1	Výroba bloku s atributy	19
3.2	Použití bloku s atributy	21
4 E	Export do SQL databáze	<u>25</u>
4.1	Konzola správy ProSQL	27
4.2	Definice tabulek	28
	Definice struktury tabulky	. 29
	Typy hodnot pro export	. 30
13	Datove typy sloupcu v SQL	. 31 31
4.4	Přiklad exportu měřičů spotřeb	33
5 P	Provoz v doméně bez lokálního uživ.	37
51	Úpravy serveru pro provoz bez uživatele	
5.1	ProCop	37
5.2	Úpravy klienta pro provoz bez uživatele ProCop	41
6 E	Data v SOL	43
6.1	Instalace SOL detabéze	<u>45</u> 44
6.2	Nastavení projektu pro ukládání dat do	45
63	SQL Správa serveru SQL	43
0.0	Obnovení, Odstranění, Změna stavu databáze	. 48
6.4	Správa databáze ProCop	49
	Automatické odstraňování alarmů	. 51
	Automatické ředění trendů	. 52
	Archivace trendů	. 53
	KEINGEXACE A SNEINK Nástroje	. 54
-		
7 L	Diagnostika zařízení	<u>57</u>
/.1 7 2	AlfaMeter	5/ 58
7.2 0 T		30
<u>ð</u> B	sezpecne nastaveni hodnoty	61
8.1 8.2	Příklad hromadného nastavení Skript hromadného nastavení hodnoty	62 65
9 A	larmy, dynamizace	<u>6</u> 7
9.1	Alarmní styly	67
9.2	Dynamizace OnAlarm	68
9.3	Zpoždění odesílání alarmů	70
10 N	Aoduly a proměnné	73

10.1 Uživatelský formát proměnných...... 75 Nastavovací dialogy76 10.2 10.3 Ukládání stavových proměnných 78 10.4 Parametry modulu PX..... 78 10.5 Výjimky časového programu v regulátoru PXC------ 84 Implementace výjimek v ProCopu 84 Index 99

© 2022 ALFA Mikrosystémy, s.r.o.

1 Moxa NPort v UDP režimu

Tato kapitola se zabývá nastavením převodníků MOXA DE-211 a DE-311 v režimu bez ovladače virtuálního sériového portu ve Windows. Převodníky Ethernet / RS232 (RS485,RS422) umožňují využít počítačové sítě k připojení zařízení standardně připojované sériovou linkou přímo k PC s monitorovacím systémem ProCop.

Pomocí převodníků je možné nahradit specializovanou galvanicky oddělenou kabeláž pro RS232, RS422, RS485 standardní počítačovou sítí. Může být využita i stávající síťová infrastruktura po domluvě se správci dané sítě, kteří musí přidělit příslušný adresní prostor pro PC a převodníky.

Å Důležité upozornění!

Nesmíme zapomenout, že narozdíl od sériové linky bude mít přenos dat přes počítačovou síť určité zpoždění, může docházet ke ztrátám části komunikace v závislosti na zatížení počítačové sítě, kompletním výpadkům sítě a v rozsáhlejších sítích i k záměně pořadí doručovaných dat. Proto NENÍ vhodné používat tento způsob propojení přes rozsáhlé sítě, či dokonce Internet u kritických průmyslových regulací. Nejlépe je volit vlastní nezávislý segment sítě.

1.1 Virtuální port vs. UDP režim

Většina převodníků Ethernet/RS232 umožňuje vyrobit virtuální sériový port přímo v operačním systému počítače. Aplikace, které umožňují využití pouze standardní sériové linky, mohou snad komunikovat i se vzdálenými zařízeními, aniž by to poznaly. Rozdílem může být pouze delší odezva a větší chybovost linky.

Problémy s virtuálními porty

Obrovským problémem při používání ovladačů virtuální sériových portů je jejich nespolehlivost při výpadcích či přetížení sítě. Při startu počítače se zavádějí ovladače virtuálních sériových portů, které navazují komunikaci s jednotlivými převodníky. Pokud se daný převodník nehlásí, virtuální port nenaváže TCP/IP spojení a sériový port je zcela nefunkční obvykle až do dalšího restartu počítače. Obdobná situace nastane, dojde-li k delšímu výpadku spojení i při provozu a komunikaci.

Důvody nenavázání, nebo rozvázání komunikace s převodníky a nefunkčnosti komunikace:

- výpadek komunikace sítě mezi PC a převodníkem
- přetížení některé části sítě mezi PC a převodníkem
- výpadek napájení některého z uzlů sítě
- výpadek napájení převodníku

Uvedené důvody i když bývají jen dočasné, přesto mohou vést k trvalému nenavázání, nebo rozvázání komunikace mezi ovladačem virtuálního sériového portu a převodníku. Opětovné navázání komunikace i když důvod výpadku již pominul, se provede až při dalším restartu PC a zavedením ovladačů.

UDP Server/Client mode

Pokud je převodník nastaven v režimu **UDP Server/Client**, data jsou přenášena samostatnými UDP pakety, které jsou odesílány přímo monitorovacím systémem ProCop bez návaznosti na ovladač virtuálního sériového portu. Pokud nastane výpadek sítě, napájení a podobně, data pochopitelně nejsou doručena. Po obnovení komunikace na úrovni sítě, jsou UDP pakety opět znovu doručovány. Počty pokusů na doručení paketů a doba čekání na odpověď je nastavitelná přímo v monitorovacím systému ProCop.

Výhody užívání režimu UDP Server/Client:

- zotavení komunikace monitorovacího systému po zotavení sítě bez restartu PC
- nastavitelný počet pokusů o doručení paketů
- nastavitelný čas čekání na odpověď na daný paket
- spolehlivější komunikace a menší zatížení PC při větším množství převodníků

8

🕭 Důležité upozornění!

Pokud používáme režim UDP Server/Client, virtuální port pro daný převodník nesmí být na PC definován! Byl-li nakonfigurován již dříve, je vhodné tento virtuální port pomocí nástrojů konfigurace virtuálních portů odstranit.

1.2 Nastavení UDP režimu

Než přistoupíme k samotnému nastavení převodníků MOXA NPort DE-311 (DE-211), budeme potřebovat statické (pevné) IP adresy jednotlivých zařízení.

Potřebné adresy:

- Dispečerské PC: IP adresa, maska sítě, výchozí brána
- Každý MOXA NPort: IP adresa, maska sítě, výchozí brána
- Komunikační UDP port (např. UDP: 4001)

🞽 Poznámka

Výchozí brány jsou potřeba jen v případě, že se jednotlivé zařízení nacházejí v různých IP sítích. Všechny IP adresy musí být unikátní, pevné, vzájemně na síti dosažitelné. Síť musí být propustná oběma směry z PC na všechny převodníky a z každého převodníku směrem na PC pro UDP pakety na zvoleném portu, implicitně 4001. Port ale může být zvolen libovolně, jak je potřeba.

b Důležité upozornění!

Pokud je na PC používán firewall, je nutné do něj nakonfigurovat výjimku pro příchozí (případně i odcházející) UDP pakety na zvoleném portu (4001). K nastavování brány firewall a dalších konfigurací systému budeme potřebovat příslušná oprávnění systému Windows.

1.2.1 Připojení k převodníku

První připojení sériovým kabelem na místě

Při prvním oživování převodníku MOXA NPort po vybalení a zapojení budeme potřebovat PC se sériovým portem a přímý kabel RS232 s příslušnými redukcemi velikosti konektorů.

Přepínače na převodníku DE-311 (DE-211) přepneme takto:

- SW1 ON
- SW2 OFF
- SW3 OFF

Připojíme přímo kabelem k sériovému portu PC. Na PC spustíme libovolný software, který umí emulovat terminály ANSI/VT100 na sériovém portu. Příkladem může být Hyperterminál, dodávaný v příslušenství pro komunikace přímo s Windows, nebo PComm Lite Terminal emulator, dodávaný přímo na CD k převodníkům MOXA NPort (bude jej potřeba nainstalovat). V nastavení emulátoru terminálu zvolíme emulaci terminálu ANSI, nebo VT100.

Otevřeme příslušný sériový port PC, nastavíme rychlost 19200,n,8,1 (19200bps, none = bez parity, 8 bitů v rámci, 1 stop bit) a NPort připojíme k napájení. Po krátké chvíli, by se měl objevit výpis volby režimu terminálu. Pokud se neobjeví, můžeme zkusit párkrát stisknout klávesu ENTER. V případě neúspěchu zkontrolujeme kabeláž, přepnutí přepínačů, odpojíme a znovu připojíme napájení převodníku, případně nastavení parametrů sériového portu PC.

襘 Tip!

Pokud je již převodník nakonfigurován jako virtuální sériový port a komunikuje, nemusíme k němu chodit, ale je možné jej překonfigurovat vzdáleně přes počítačovou síť.

Vzdálené připojení přes síť, byl-li převodník již dříve konfigurován

Pokud jsme již měli dříve převodník nakonfigurován v síti jako virtuální sériový port, nemusíme k němu chodit s kabelem. Otevřeme správu virtuálních sériových portů, zjistíme poznačíme si jeho IP adresu, síťovou masku, případně výchozí bránu a virtuální sériový port odstraníme.

Å Upozornění!

Běží-li na PC monitorovací systém, bude nutné přejít do režimu designer, vybrat RS232 cestu příslušného sériového portu a v lokální nabídce, kterou lze otevřít pravým tlačítkem myši zvolit "Zastavit podřízené".

Nyní můžeme přejít k připojení k vybranému převodníku. Spustíme konzolu příkazového řádku. V nabídce Start, v

řádku Spustit napíšeme "**cmd**" a stiskneme ENTER, nebo zvolíme Start\Všechny program\Příslušenství \Příkazový řádek. Otevře se okno s obvykle černým pozadím.

Napíšeme příkaz:

telnet 192.168.1.240

kde uvedeme správnou IP adresu převodníku získanou z konfigurace virtuálních sériových portů a stiskneme ENTER.

Příkazový řádek a spuštění emulátoru terminálu telnet po síti na IP adresu převodníku



Pokud je převodník dostupný pro vzdálenou správu, měl by se objevit dotaz na typ terminálu, obdobně jako při přímém připojení přes sériový kabel.

🕭 Upozornění!

Pokud je SW1 přepnut do polohy ON, nelze se připojit terminálem telnet přes počítačovou síť, jen přes sériovou linku přímo u zařízení. Toto platí i opačně, pokud je SW1 v poloze OFF, nelze se připojit přes sériovou linku, pouze vzdáleně přes počítačovou síť.

Volba terminálu po připojení (RS232 i telnet)

Po připojení emulátorem terminálu na sériové lince RS232 (hyperterminal, PComm Lite), nebo přes počítačovou síť (telnet) se zobrazí výpis volby terminálu.



Telnet 192.168.1.240	
Console terminal type (1: ansi/vt100, 2: vt52) : 1	.

Ponecháme volbu na 1 a stiskneme ENTER.

🕭 Upozornění!

Bylo-li dříve do převodníku zadáno heslo pro přístup na konfiguraci, bude nyní vyžadováno. Zadáme jej a stiskneme ENTER.

1.2.2 Konfigurace IP adresy

Nejprve poznamenejme, že další kapitoly jsou významově téměř shodné pro konfiguraci přes sériovou linku i síťový terminál telnet. M ohou se lišit zejména v designu zobrazovaných oken dle zvoleného terminálu, nikoliv však významově. Případné významové odlišnosti budou zvláště poznamenány.

Hlavní nabídka a ovládání

Po připojení ⁸, volbě terminálu a případném zadání hesla by se měla zobrazit hlavní nabídka konfigurace převodníku NPort. Po nabídce se pohybujeme šipkami <u>Vlevo</u> a <u>Vpravo</u>, volbu provedeme klávesou <u>ENTER</u>, návrat klávesou <u>ESC</u>. Pro přechod mezi zadávacími kolonkami je možné použít klávesu <u>TAB</u>, šipky Nahoru a Dolů, v kolonce pak <u>Vlevo</u> a <u>Vpravo</u>. Výběr ze seznamu, nebo otevření další podnabídky se provádí klávesou <u>ENTER</u>. Úpravy hodnot se provádějí přepisem, klávesy <u>BackSpace</u> a <u>Delete</u> provádějí vymazání předchozího znaku. Znak je možné přepsat klávesou <u>Space</u> (mezerou) na prázdný. Po hlavní nabídce se můžeme pohybovat i volbou zvětšeného písmena.

📥 Důležité upozornění!

Veškeré úpravy v konfiguraci se projeví až po uložení a restartu zařízení. Pro zapsání konfigurace zvolíme v hlavní

Tipy a triky

nabídce Restart a potvrdíme uložení konfigurace klávesou ENTER.

💢 Tip!

Pokud **nechceme** změny v konfiguraci uložit, zvolíme z hlavní nabídky volbu Exit. Případný dotaz na uložení hodnot dementujeme klávesou ESC.

Konfigurace IP Adresy

V hlavní nabídce zvolíme volbu serverConfig.

Konfigurace IP adresy převodníku

Can Telnet 192.168.1.240				
MOXA NPort Express V2.6				
[serverConfig] OP_mode Serialpor Config server settings	t Monitor Ping Rest	art Exit		
ESC: back to menu Enter: selec	t			
Server Model Server Name Serial Number IP Configuration	DE-311 Rev.2 [Jmeno naseho prevodn 57288 [Static IP]	iu I		
IP Hddress Netmask Gateway Ethernet Status MAC Address	[192.168.1.240] [255.255.255.0] [192.168.1.1] 100M/Link 00:90:E8:04:DF:C8			
Password	[]	-		

Převodník si můžeme pojmenovat, abychom ho kdykoliv identifikovali. Jméno je viditelné jen v konfiguraci a může být libovolné.

🞽 Poznámka

Pokud jsme připojeni přes síť emulátorem terminálu telnet, pravděpodobně již máme správně nastavenu IP adresu. Pokud ji však změníme, promítne se tato změna až při ukončení konfigurace. Nové připojení již budeme muset provést na novou IP adresu. Pokud byla adresa zadána chybně, nebo koliduje s jiným zařízením, budeme muset k převodníku dojít a nastavit správně přes sériovou linku 8

Nastavíme IP adresu. Zvolíme z nabídky Static IP a zadáme IP adresu, masku Netmask, případně výchozí bránu Gateway. Pro zvýšení bezpečnosti můžeme zadat heslo Password pro přístup ke konfiguraci převodníku.

Pokud je vše správně nastaveno, můžeme z nabídky odejít klávesou ESC.

1.2.3 Konfigurace UDP režimu

V hlavní nabídce zvolíme volbu OP_mode. V levé spodní části Application vyberme z nabídky UDP Server/ Client. Výběr režimu převodníku



Dále bude třeba nastavit parametry přenosu UDP paketů mezi PC dispečinku a převodníkem NPort. V pravé části zvolíme More setting, Select for more setting.

Přístup k nastavení parametrů UDP režimu převodníku



Následující tabulka zobrazuje parametry, které je potřeba nastavit pro PC dispečinku s IP adresou 192.168.1.11. Pokud má PC adresu jinou, zadáme ji místo 192.168.1.11 na všechna čtyři místa. Dále je zde nastaven komunikační UDP port 4001 pro oba směry komunikace. Pokud je potřeba komunikovat po jiném portu, přepíšeme jej ve všech kolonkách.

💢 Tip!

Pokud chceme jednotlivá zařízení rozčlenit podle převodníků, každému z nich nastavíme jiný UDP port, např. 4001, 4002, 4003... a to v obou směrech. To znamená, že při konfiguraci jednoho převodníku použijeme ve všech případech i stejný port, v prvním všude 4001, v druhém všude 4002. Každý port pak bude znamenat vložení nové cesty NPortWay v projektu Monitorovacího systému ProCop.

Dále nastavíme maximální čas pro odeslání paketu Force transmit (ms). Tento čas udává, po jaké době bude automaticky paket s daty ze sériové linky odeslán. Tento parametr bývá nastaven na 0. Pro různá zařízení je potřeba vyzkoušet, jaký čas bude vhodný. Například pro MBus je vhodné zvolit čas okolo 300ms.

Adresy pro přenos dat v UDP režimu, UDP porty a odesílací čas

. Telnet 192.168.1.240			- • ×
MOXA NPort E	xpress V2.6		^
serverConfig <mark>[OP_mode]</mark> Serialport Moni Setup OP mode	tor Ping Restart	Exit	
ESC: back to menu Enter: select			
Port Appl Begin 01 [UDP] Serial to Ethernet Dest. IP 1 [192.168.1.11	End][192.168.1.11	Port][4001]][4001]]
Dest. IP 3 [Dest. IP 4 [Ethernet_to_serial <optional< th=""><th></th><th>][4001]][4001]</th><th></th></optional<>][4001]][4001]	
Src. IP 1 [192.168.1.11 Src. IP 2 [Src. IP 3 [Src. IP 4 [J[192.168.1.11][][][][]]]	
Data packing(optional)	ີ [4001]		
+ Force transmit (ms):	[300]]		+ *

Pro vysvětlení uveďme, že se částí Serial to Ethernet konfiguruje, na jaké až 4 rozsahy IP adres mají být zaslány UDP pakety s definovaným Portem. V našem případě je definován jediný počítač - rozsah právě jedné adresy dispečinku. Část Ethernet to serial (optional) definuje, ze kterých počítačů se data příchozí na Local listen Port přepošlou na sériovou linku. V našem případě se jedná opět o jediný a to dispečerský počítač. UDP Port nastavíme později v NPort cestě monitorovacího systému ProCop.

Pokud je vše správně nastaveno, můžeme z nabídky odejít klávesou ESC.

1.2.4 Nastavení sériového portu

Narozdíl od režimu virtuálního portu, kde se komunikační parametry sériové linky zadávaly v příslušné cestě Monitorovacího systému ProCop a přenášely při komunikaci, nyní musíme zadat komunikační parametry přímo do převodníku.

💢 Tip!

Pokud předěláváme stávající virtuální port na UDP režim, je vhodné se podívat do Panelu nástrojů ProCop v režimu Designer a ve správci projektu najít příslušnou nahrazovanou cestu RS232, původně připojenou k virtuálnímu portu a komunikační parametry převzít přímo z ní. Parametry nastavované v konfigurátoru virtuálních portů se pro komunikaci Monitorovacím systémem ProCop neuplatňují.

Nastavení komunikačních parametrů sériové linky

V hlavní nabídce zvolíme volbu Serialport. V kolonce Baud Rate(bps) nastavíme komunikační rychlost, v dalších kolonkách pak paritu, délku rámce, počet stop bitů, řízení toku a používání vyrovnávací paměti na sériovém portu. Nastavení komunikačních parametrů pro M-Bus

Telnet 192.168.1.240				- D X
1	10XA NPort Expre	ss V2.6		·
serverConfig OP_mode [Seria Config serial port settings	alport] Monitor	Ping Restart	Exit	
ESC: back to menu Enter: s	select			
Port Number	1			
Baud Rate(bps)	2400			
Data Bit				
Stop Bit Flow Control	[1] [None]			
UART FIFO	[Enable]			
				-

V obrázku jsou nastaveny komunikační parametry obvyklé pro převodník RS232/M-Bus, 2400,e,8,1.

Pokud je vše správně nastaveno, můžeme z nabídky odejít klávesou ESC

12

1.2.5 Uložení konfigurace

Pokud jsme vše nastavili správně, můžeme v hlavní nabídce zvolit volbu Exit. Je zobrazena otázka, zda chceme data uložit do EEPROM a převodník restartovat. Pokud jsme si nastavením jisti, stiskneme klávesu ENTER pro potvrzení. Pokud chceme všechny změny stornovat, stiskneme klávesu ESC.

Telnet 192.168.1.240	
MOXA NPort Express V2.6	^
serverConfig OP_mode Serialport Monitor Ping Restart [Exit] Quit the session	
ESC: back to menu Enter: select	
The configuration has been changed. Please press <enter> to reboot for new configuration or press <esc> to cancel.</esc></enter>	
WARNING!! Restart system will disconnect all ports	
	-

Uložení změn nastavení a restart převodníku

Å Důležité upozornění!

Nastavení ukládáme jen tehdy, když jsme si jisti, že je nastavení IP adresy zařízení správné, nebo jsme je nezměnili. Ostatní nastavení jako OP_mode či Serialport lze kdykoliv vzdáleně změnit, ne však, pokud se do zařízení již vzdáleně nedostaneme. V takovém případě můžeme kdykoliv zařízení nakonfigurovat přímo přes port RS232⁸ na místě. Pokud jsme nastavili **heslo**, bude nadále vyžadováno pro konfiguraci! Raději si jej poznamenejme.

1.3 Úpravy projektu pro N-Port

Nyní bude potřeba upravit monitorovací projekt. Narozdíl od virtuálních sériových portů není potřeba mít zvlášť jednotlivé cesty. Všechny převodníky, které mají nastaven shodný UDP port (v našem příkladu 4001) stačí jediná komunikační cesta pro všechna zařízení.

💢 Tip!

Pokud chceme jednotlivá zařízení rozčlenit podle převodníků, každému z nich nastavíme jiný UDP port 1, např. 4001, 4002, 4003... Pak bude prokaždý původní virtuální sériový port - pro jeden převodník - právě jedna komunikační cesta NPortWay.

Vložení a nastavení komunikační cesty NPortWay

V Panelu nástrojů ProCop otevřeme v režimu Designer Správu projektu, vybereme Cesty a moduly a stiskneme klávesu Insert, nebo symbol zeleného plus. Vybereme NPortWay. Cestu pojmenujeme.

Vložení komunikační cesty NPortWay

Uložit	×
⊂Paramet Typ:	ry [NPort₩ay
Info:	Cesta pro komunikaci Moxa NPort 5xxx, DE-211, DE-311, UDP Raw (Client/Server)
Verze:	3.00.00
	<u> </u>

Ve vlastnostech a dynamizacích cesty je adresa PC (0.0.0.0), tu můžeme ponechat, sama se při spuštění načte. V části

Síť, v položce UDP port nastavíme příslušný port, který jsme nastavili do převodníku. V našem případě 4001. Položka Maximálně spojů určuje, kolik zařízení může společně komunikovat současně. Pro zařízení MBus necháme počet na 1.

V části Zpozdit potvrzení odeslání dle rychlosti, v položce Rychlost jen pro výpočet zpoždění uvedeme rychlost komunikace na portu RS232. V našem případě 2400bps. Tato rychlost slouží jen pro výpočet předpokládaného komunikačního zpoždění. Musí být správně nakonfigurována i u sériového portu v převodníku 12.

🔢 Vlastnosti a dynamizace		
	Hodnota	
🖩 Název	NPort_1	
Popis		
🗄 0 cestě	NPortWay	
Adresa (X.X.X.X)	0.0.0.0	
💷 Spojení zpět (IP, tel. čísl		
🗆 Síť		
UDP port	4001	
🔲 Maximálně spojů	1	
Zpozdit potvrzení ode		
🔲 Rychlost jen pro výpo	2400 bps	
🔲 Přičíst na paket [q]	0 s	

Parametry komunikační cesty NPortWay pro MBus.

Nyní můžeme vložit nové moduly, nebo stávající moduly přesunout z cesty RS232 pod cestu NPortWay. Při přesouvání označíme ve stromu cest a modulů cestu, v pravé částí označíme všechny moduly zvolíme Úpravy/Vyjmout, nebo klávesovou zkratkou Ctrl+X. Vybereme cestu NPortWay a zvolíme Úpravy/Vložit, nebo stiskneme klávesy Ctrl+V. Označíme moduly, které komunikují přes stejný převodník NPort a zvolíme Vlastnosti a dynamizace Ctrl+ENTER. V Parametrech spojení, položce Tel. číslo, adresa... zadáme IP adresu převodníku NPort, přes který jsou dané zařízení připojeny. Můžeme zvýšit Počet opakování v parametrech komunikace, například na 10 pokusů.

IP adresa NPort převodníku v modulu M-Bus.

🔢 Vlastnosti a dynamizace		
	Hodnota	
💷 Název	MTx	
💷 Prefix	MTX	
🗉 Popis		
Soubor	MTx.pio	
🗄 O modulu	MBusDeviceModule	
Parametry komunika		
Adresa (X.X)	0.1	
💷 Permanentní obnova	1h	
💷 Počet opakování	10	
💷 Čekat na odpověď	3 s	
🗆 Perioda obnovy		
🖩 Perioda (0=vyp)	5 s	
💷 Posunutí periody	0 s	
💷 Obnov při startu	<ne></ne>	
🔲 Validita (0=vždy)	0 s	
💷 První obnova (0=o	0 s	
Poslední obnova (1	1 d	
🗆 Parametry spojení		
💷 Pokusů pro navázání	3	
🗉 Tel. číslo, adresa	192.168.1.240	
💷 Komunikační oblast		
Parametry M-Bus		
Typ zařízení (autokonfig	AEW 21.2: Objem [m3]	

V jedné NPort cestě mohou být moduly, které jsou za různými zařízeními. IP adresy v parametru spojení musí odpovídat danému NPortu. Logické adresy (např. M-Bus) zůstávají beze změny.

🗾 Poznámka

Pokud jsme přesouvali moduly z cesty RS232 a je cesta již prázdná, můžeme ji zrušit.

1.4 Problémy s komunikací

Po spuštění projektu by měla komunikace se zařízeními fungovat. Uveď me pár případů problémů, které mohou nastat a jak je řešit.

Zařízení nekomunikují

Nejprve zkusíme provést test, zda je NPort po síti z PC dosažitelný.

Spustíme konzolu příkazového řádku. V nabídce Start, v řádku Spustit napíšeme "**cmd**" a stiskneme **ENTER**, nebo zvolíme Start\Všechny program\Příslušenství\Příkazový řádek. Otevře se okno s obvykle černým pozadím. napíšeme příkaz:

ping 192.168.1.240

kde použijeme místo 192.168.1.240 IP adresu daného převodníku. Pokud převodník odpovídá, vypisují se časy odezvy paketů. Převodník je tedy na síti viditelný.

Problém může být v nastavení Windows Firewall. Otevřeme v Ovládacích panelech jeho správu a definujeme výjimky pro všechny použité UDP porty např. 4001, 4002 ... Zkontrolujeme, že jsou výjimky na firewallu povoleny.

Druhý problém může být chybně konfigurovanou komunikační cestou - zkontrolujeme podle nastavení 13).

Také může být nevyplněna, nebo chybně vyplněna IP adresa NPort převodníku v komunikačním modulu, zkontrolujeme

Dále můžou být chybně nastaveny parametry (IP adresy, UDP port 11) v převodníku NPort, nebo komunikační parametry na sériovém portu 12 NPort.

Pokud je vše v pořádku, je pravděpodobné, že je problém za NPortem na lince RS232.

Zařízení komunikují, ale objevují se komunikační chyby

Zkontrolujeme parametry komunikačních modulů 13 v Panelu nástrojů ProCop, zejména Počet opakování a Čekání na odpověď.

Nebo je síť nespolehlivá, přetížená a podobně.

2 Automatické vlastnosti potrubí

ProCop verze 3.3 disponuje možností nastavit automatickou barvu potrubí (i vzduchotechnického). Potrubí se vykresluje šedě, pokud není napojeno na potrubí jiné, než automatické barvy.

🗆 Vlastnosti potrubí 👘		
🖪 Barva	<automatická></automatická>	<u> </u>
💷 Šířka	7 bodů	•

Nastavení automatické barvy potrubí

Následující obrázek ve třech krocích zobrazuje, k čemu je možné použít automatické barvy potrubí. V celé síti trubek stačí jediná s danou barvou, například přívodní. Po připojení na ostatní potrubí, všechny navázané s automatickou barvou kopírují barvu připojeného potrubí. Změníme-li barvu jediného potrubí, změní se barva i ostatních připojených.

Označená trubka s pevnou barvou, ostatní automatické, napojení a změna barvy



Napojování potrubí do skupin a knihovních bloků

Výše popsaná vlastnost se hlavně uplatní v kombinaci s knihovními bloky a skupinami. Vložíme-li knihovní blok, nebo vytvoříme-li skupinu s příznakem Spojovat automaticky, potrubí se napojí i v rámci knihovního bloku a skupiny, případně přebere barvu potrubí.

Kninovni biok, nebo skupina s priznakem automatickeno spojov
--

🗆 Příznaky	
Zmrazit entitu	Ne>
Zakázat výběr myší	Ne>
Netisknout	Ne>
Zakázat rotaci	Ne>
Zakázat export	Ne>
Spojovat automaticky	Ano>
💷 Úroveň přístupu	Trivial
Perioda obnovy [ms]	1000

Podmínkou je automatická barva potrubí v knihovním bloku či skupině a příznak Spojovat automaticky. Posuneme-li knihovní blok, či skupinu do blízkosti potrubí, automaticky se napojí a změní svou barvu.

Napojení knihovního bloku na potrubí



3 Atributy knihovních bloků

Knihovní bloky jsou již od ProCop verze 3.2 vybaveny funkcí pro definici atributů. Ve verzi 3.3 byly tyto vlastnosti dle zkušeností dále upraveny, aby se lépe a jednodušeji používaly

K čemu atributy slouží

Knihovní bloky po vložení do displeje bylo nutné obvykle oddělit a jednotlivé dynamizované části napojit na požadované proměnné. Nevýhodou je pracnost a odpojení od knihovny. Při úpravě takovéhoto bloku v knihovně se změny neprojeví v displejích, jelikož jsou bloky od knihovny odpojeny.

Atributy se snaží tento problém vyřešit. Bloku v knihovně se nadefinují virtuální proměnné - atributy - které se pak v editoru pouze napojí. Nedojde tak k odpojení z knihovny a případné úpravy bloku v knihovně se projeví ve všech instancích v displejích. Navíc je možné atributy vyrobit tak, aby svými jmény přispěly k automatickému napojení ostatních proměnných. K tomuto napojení v displeji slouží Průvodce napojení atributů Ctrl+Q.

Barevné významy úchopových bodů

Pro informaci o napojení bloků do knihovny se úchopové body body vykreslují modře. Zvětšení úchopových bodů, případně červená barva znázorňuje, že je entita dynamizována a obsahuje odkazy na technologické proměnné.

Možné barvy a velikosti úchopových bodů: modrá = napojený knihovní blok, zvětšení = dynamizovaná entita



3.1 Výroba bloku s atributy

Pro snadné pochopení bude nejlepší vyrobit knihovní blok s atributy a pak jej použít v displeji. Vybereme nějaký kus z existujícího projektu, nejlépe takový, který se více krát opakuje, ale je pokaždé napojen na různé proměnné. Jako příklad bude použita měřená a žádaná teplota náběhu ve směšovacím okruhu MxCrt v regulátoru PX. Nakopírujeme si jej do schránky Ctrl+C.

Nejprve otevřeme ve stromu Knihovník projektovou knihovnu Project Library. Přidáme knihovní blok například tlačítkem Insert, pojmenujeme jej a otevřeme.

Definice atributů

Vložíme obsah schránky do knihovního bloku a zvolíme Úpravy/Převod na atributy. Nezapomeneme vybrat případná napojovaná potrubí a změníme jim barvu na automatickou. Celému knihovnímu bloku ve stromu knihovníka nastavíme příznak Spojovat automaticky, je-li to potřeba.

Knihovní blok po převodu na atributy s automatickou barvou trubky



V nástrojové liště Vlastnosti zvolíme Upravit vlastnosti displeje (symbol modrého klíče) a ve vlastnostech a dynamizacích zobrazíme Tabulku atributů. V tabulce atributů nyní vidíme seznam jednotlivých automaticky vytvořených atributů.

Každý atribut má tyto vlastnosti:

- Název jméno atributu, v dynamizacích bloku se použije místo proměnné, na začátku doplněný o znak \$
- Popis textový popis atributu, zobrazovaný při napojování knihovního bloku v průvodci a vlastnostech, měl by přesně definovat význam proměnné
- Typ požadovaný významový typ atributu pro výběr a zadání slouží tlačítko se třemi tečkami atributu smí být přiřazena jen:
 - o Proměnná/Hodnota technologická proměnná, nebo konstantní hodnota
 - o Proměnná technologická proměnná, sufix
 - o Výraz libovolný výraz
 - Text libovolný text průvodce vždy vyžaduje nové vyplnění, nebo potvrzení stávajícího textu
 - **Text implicitní** libovolný text průvodce nevyžaduje vyplnění, vždy použije předvolbu, ale je text možno změnit
 - Text skládaný libovolný text průvodce vždy vyžaduje vyplnění, ale jen jednou pro všechny vnořené bloky s atributem stejného jména
 - Hodnota lze zadat jen konstantu
 - Seznam v editoru zadáme seznam zobrazovaných názvů a reálných hodnot atributu, vhodné pro výběr z entit
 - Zaškrtávácí políčko v editoru zadáme popis a hodnotu pro zaškrtnuté a prázdné políčko, vhodné např. pro viditelnost entity
- Datový typ konkrétní datový typ atributu, tedy přiřazené proměnné, výrazu, textu, nebo jakýkoliv typ
- **Předvolba** implicitně nabízená předvolba hodnoty atributu, prázdná u hlavního atributu, odkazy na příznaky jiných atributů, implicitní text...
- Hodnota nepoužití pokud může být atribut napojen, zde se uvede hodnota, která daný atribut vyřadí tak, aby se v bloku neprojevoval

Tabulka	atributů	no	automatickém	nřevodu	atributů
1 aouna	an would	P°	antomationent	prevouu	an womm

🕼 Tabulka atribu	tů					
📄 🐹 💼 🛙 🖻	i 🔶 🂠 💥 🕅 🥼 📾					
Název	Popis	Тур	Datový typ	Předvolba	Hodnota nepoužit	í
TXT_Popis	Popis Teplota náběhu	Text impli 🔳	Text	Teplota náběhu	<u> </u>	••
HysFrPrt	Hystereze protizamrazu	Proměnná 🔳	Analog			-
HysFrPrt_Ack	Hystereze protizamrazu - Ack	Proměnná 🔳	Binary	\$HysFrPrt.Ack	■ 0 B	•
HysFrPrt_Status	Hystereze protizamrazu - Status	Proměnná 🔳	Discrete	\$HysFrPrt.Status		-
SpDtr	Určení žádané hodnoty	Proměnná 🔳	Analog		<u> </u>	-
TFI	Teplota náběhu	Proměnná 🔳	Analog		<u> </u>	-
TFI_Ack	Teplota náběhu - Ack	Proměnná 🔳	Binary	🔳 \$TFI.Ack		-
TFI_Reliability	Teplota náběhu - Reliability	Proměnná 🔳	Discrete	\$TFI.Reliability		-
TFI_Status	Teplota náběhu - Status	Proměnná 🔳	Discrete	\$TFI.Status	0 1	-
					✓ Zavřít	

Názvy atributů se používají místo proměnných, nebo jejich sufixů v dynamizacích, hodnotách či textech entit. Název uživatelského atributu se uvozuje znakem \$ na začátku, například \$TXT_Popis.

21

Systémové atributy

Vyjma uživatelských atributů jsou k dispozici **předdefinované systémové atributy**. Tyto atributy se mohou použít přímo v dynamizacích, či textech knihovních bloků, nebo rovněž v předvolbách definice uživatelských atributů. Tyto atributy jsou nabízeny v rozbalovacích seznamech (při vkládání proměnné, předvolby atributu příslušného typu) a jsou z obou stran uvozeny znaky \$sys.att\$.

V současnosti jsou k dispozici tyto systémové atributy:

- **\$Display.Name\$** jméno technologického displeje, ve kterém je blok vložen
- **\$Display.Descr\$** pop is technologického displeje, ve kterém je blok vložen
- **\$Display.Title\$** popis, nebo jméno technologického displeje, je-li popis prázdný
- \$Visual.Name\$ jméno vizuálního projektu
- **\$Visual.Descr\$** popis vizuálního projektu
- **\$Visual.Title\$** popis, nebo jméno vizuálního projektu, je-li popis prázdný
- \$Project.Name\$ jméno datového projektu
- **\$Project.Descr\$** popis datového projektu
- **\$Project.Title\$** popis, nebo jméno datového projektu, je-li popis prázdný

🗾 Poznámka

Je vhodné používat systémové proměnné \$. Title\$, neboť máme zaručeno, že není-li zadán popis, bude doplněno jméno dle daného atributu.

Použití atributů v dynamizacích knihovního bloku

Máme-li nadefinovány atributy knihovního bloku, nebo automaticky vygenerovány příkazem Úpravy/Převod na atributy, můžeme přejít k tvorbě a úpravám dynamizací jednotlivých entit. Převod na atributy automaticky provede nejen vytvoření atributů, ale i úpravy dynamizací všech entit knihovního bloku. Dynamizace je však vhodné zkontrolovat.

Pokud vytváříme dynamizace nové, používáme atributy, či systémové atributy shodně, jako by se jednalo o proměnné. V dialogovém okně pro Výběr proměnné, funkce a výrazu lze v rozbalovacím seznamu Proměnná nebo výraz vybrat příslušný atribut.

Pro použití atributů zejména typu Proměnná/Hodnota byly definovány dvě nové funkce jazyka Bára:

Funkce	Popis
VarExist(variable) of binary	Vrací TRUE, pokud proměnná existuje
VarOrConst(ident) of binary	Vrací TRUE, je-li parametr proměnná či konstanta

Funkce VarExist vrací logickou hodnotu existence proměnné. Pokud byla do atributu typu Proměnná/Hodnota zadána hodnota (konstanta, např. z parametru hodnota nepoužití), je možné tuto skutečnost v dynamizacích ošetřit, například v lokální nabídce podmínkou viditelnosti daného řádku, v přiřazení hodnoty podmínkou pro provedení přiřazení a podobně, dynamizovat na viditelnost entity dle výsledku funkce a podobně. Obdobně lze využít i funkci VarOrConst pro otestování, zda byla proměnná, nebo hodnota vůbec do atributu zadána.

3.2 Použití bloku s atributy

Máme-li vyroben knihovní blok s atributy, můžeme jej zkusit použít v projektu a napojit. Případně zvolíme blok dodávaný v knihovně monitorovacího systému ProCop.

Otevřeme příslušný technologický displej a zvolíme Kreslení/Vložit blok z knihovny klávesa \mathbf{k} . Vybereme námi vyrobené nebo dodávaný knihovní blok. Vložíme jej, umístíme jej kam potřebujeme a zkontrolujeme, že se nám potrubí propojilo a má správnou barvu. Pokud ne, blok nemá v knihovně nastaven příznak Spojovat automaticky, nebo vnořený blok knihovního bloku nemá tento parametr nastaven. Případně nemá potrubí nastavenu **automatickou barvu**.

Knihovní blok s atributy po vložení a napojení na technologii



Průvodce napojení atributů

Po vložení máme dvě možnosti, jak s knihovním blokem s atributy pracovat. Jednodušší je spustit z nabídky Úpravy/ Průvodce napojení atributů Ctrl+Q.

Pokud je knihovní blok nově vložen, nejprve se objeví dotaz na Výběr prefixu a skupiny.

Výběr prefixu skupiny proměnných pro napojení atributu



Po regulátoru a skupiny proměnných, na které budeme dále atributy napojovat se automaticky spustí propojování atributů. K zadanému **prefixu** skupiny se připojí postupně **jméno** všech atributů, které byly zadány v knihovním bloku. **Existuje-li proměnná**, vzniklá **spojením vybraného prefixu a jména atributu** (bez znaku \$), **automaticky se proměnná napojí na atribut** a pokračuje se dalším atributem.

Pokud není proměnná nalezena, automatické napojování se zastaví a čeká se na zásah obsluhy.

Jsou k dispozici tyto možnosti:

- požadovaná proměnná se jmenuje jinak, nebo je z jiné skupiny je potřeba zadat či vybrat správnou proměnnou
- daný vložený blok je jiného typu, je možné Vybrat jiný blok, byl-li definován (např. 1 či 2 stupňový ventilátor, ...)
- daný podblok není implementován, můžeme jej celý Skrýt
- daná proměnná neexistuje, tlačítkem Nepoužito vložíme Hodnotu nepoužití
- pokud nic nevyhovuje, musím ukončit průvodce, blok oddělit a upravit dynamizovat pole potřeb, již napojené atributy se přepíší proměnnými

🔽 Průvodce	
Teplota náběhu ???? ???? max min total	 Attributy Popis Teplota náběhu [TXT_Popis] A Hystereze protizamrazu [HysFrPrt] B Hystereze protizamrazu - Ack [HysFrPrt_Ack] D Hystereze protizamrazu - Status [HysFrPrt_Status] A Určení žádané hodnoty [SpDtr] A Teplota náběhu [TFI] B Teplota náběhu - Ack [TFI_Ack] D Teplota náběhu - Ack [TFI_Ack] D Teplota náběhu - Ack [TFI_Ack] D Teplota náběhu - Status [TFI_Status] Zadejte proměnnou pro atribut typu: Analog Název: Hystereze protizamrazu Nézev: Hystereze protizamrazu Nebo vyberte jiný blok: Blok: Attributy Attributy Despoužito Proměnná © Použít
6/9	🗙 Skrýt 💽 🌩 Další 🗸 OK

V dané implementaci nepoužitá proměnná, problém vyřešíme tlačítkem nepoužito

Pokud byly napojeny všechny proměnné, přepsány všechno požadované texty a vyřešeny všechny problémy, průvodce úspěšné napojení ohlásí. Procházením po stromu atributů můžeme zkontrolovat jejich napojení, v nejvyšší úrovni lze **změnit celý prefix** napojování bloku.

🕼 Průvodce	
Teplota náběhu ???? ???? max min ?	Attributy Popis Teplota náběhu [TXT_Popis] A Hystereze protizamrazu [HysFrPrt] Hystereze protizamrazu - Ack [HysFrPrt_Ack] Hystereze protizamrazu - Status [HysFrPrt_Status] A Určení žádané hodnoty [SpDt] A Teplota náběhu [TFI] Teplota náběhu - Ack [TFI_Ack] Teplota náběhu - Reliability [TFI_Reliability] Teplota náběhu - Status [TFI_Status] Attributy Hotovo, vše vyřešeno.
9/9	🗙 Skrýt 🕒 Další 🗸 OK



Ruční úpravy napojení atributů

Druhou variantou úprav je ruční napojení jednotlivých atributů ve Vlastnostech a dynamizacích klávesy \sim , Ctrl +Enter daného knihovního bloku. Napojíme-li jeden atribut, všechny ostatní nenapojené se stejným algoritmem jako v průvodci pokusí napojit na proměnné.

4 Export do SQL databáze

Export do SQL databáze

ProCop od verze 3.4 disponuje možností automatického exportu dat do SQL databází, XLS, XLSX, CSV a jiných formátů, dostupných přes ODBC ovladače Windows. Pro tyto exporty je v monitorovacím systému služba ProSQL, která v naplánovaných časech spouští definované skripty, které vytvářejí, doplňují, nebo modifikují databázi, nebo soubory. Tento návod popisuje definici exportu dat Microsoft SQL databáze.

Požadavky před realizací

Předpokládejme fungující, běžící monitorovací systém ProCop verze 3.4 a vyšší, který sbírá data z regulátorů, či měřičů spotřeb po libovolných komunikačních cestách. Tato data, nebo některá z nich je potřeba exportovat do SQL databáze, například pro nadřízené systémy jako jsou vyhodnocovací, fakturační či firemní informační systémy. Tyto systémy obvykle pracují nad SQL databází, do které bude potřebné v definovaném formátu dodávat data z monitorovacího systému. Může být také vytvořena specializovaná SQL databáze, která bude sloužit pouze pro výměnu dat mezi monitorovacím systémem ProCop a informačním systémem.

Co je potřeba před realizací exportu:

- Funkční SQL databáze
- Název SQL databáze, případně instance
- IP adresa, nebo doménové jméno počítače, na kterém SQL databáze běží
- Uživatelské jméno a heslo pro čtení i zápis do SQL databáze
- Monitorovací systém ProCop v. 3.4 a vyšší, sbírající data z technologie
- Funkční síťové spojení mezi oběma počítači, běží-li SQL databáze na jiném počítači
- Struktura, definice tabulek a formát dat, v jaké podobě mají být data ukládána

Instalace Microsoft SQL Server 2008 R2

Instalaci databázového serveru, pokud nepoužíváme existující provedeme podle dokumentace výrobce. V případě použití Microsoft SQL Server 2008 R2 Express Edition je instalace velmi snadná. Stačí stáhnout (v současnosti ze stránek www.microsoft.com/sqlserver/) příslušný softwarový balík a nainstalovat do Windows Server 2008 R2, případně Windows 7 Professional. Podle současných licenčních podmínek je možné využívat databázový server ve verzi Express do 10GB dat zcela zdarma.

b Důležité upozornění!

Licence SQL databází nejsou součástí monitorovacího systému ProCop a řídí se licenčními podmínkami a pravidly konkrétního výrobce.

Uveď me krátký přehled a stručné vysvětlení dále používaného názvosloví, které se při instalaci SQL serveru a při realizaci projektu ProSQL využívá:

- SQL server (SW) softwarový balík, který umožňuje zpracovávat a uchovávat data, přístupný přes SQL (Structured Query Language = strukturovaný dotazovací jazyk), někdy se také hovoří o SQL databázi
- SQL server (HW) počítač, na kterém je provozována SQL databáze
- Instance (SQL serveru) na jednom počítači může být nainstalováno více SQL serverů, tedy více instancí tohoto softwaru, rozlišují se jménem instance
- Databáze v SQL serveru může být více databází, nezávislých úložišť dat, každá s vlastní strukturou a oprávněními

26

- Tabulka databáze se mimo jiné skládá z tabulek, které mají definovány sloupce a v řádcích se pak ukládají data
- Sloupec vertikální část tabulky, definovaná jménem a datovým typem (případně dalšími parametry), jméno sloupce musí být v rámci tabulky unikátní, nikoliv v rámci databáze; v řádcích se pak v daném sloupci ukládají data definovaného jména a typu
- Řádek horizontální část tabuky, kde jsou ukládána data všech, nebo některých sloupců
- Buňka průnik řádku a sloupce tabulky, hodnota dat daného řádku sloupce
- Index podpůrný mechanismus pro rychlé třídění a vyhledávání v tabulce, obvykle pro daný sloupec
- **Primární index** index sloupce, kde všechna data musí být vyplněna a musí být unikátní (např. pořadové číslo)
- Složený index index, který je vytvořen nad více sloupci
- Struktura databáze definice struktur všech tabulek, jejich jmen a struktur v dané databázi
- Struktura tabulky definice všech sloupců tabulky, jejich jmen, datových typů a dalších parametrů, indexů a vazeb a omezení
- Struktura dat definice obsahu řádků, jejich identifikace

🚖 Tip!

Pro správu Microsoft SQL databází je dodáván produkt SQL Server Management Studio, což je velice silný nástroj pro veškerou správu databází a to i vzdáleně.

Struktura databáze

Ve stručnosti uveďme, že se jedná o definici tabulek, jejich sloupců, indexů, formátů dat, časů a period ukládání ve kterém mají být data z monitorovacího systému ProCop exportována. Samotnou strukturu tabulek bude nutné nadefinovat v monitorovacím systému ProCop, který ji sám vytvoří v SQL databázi po spuštění exportních úloh. Je potřeba zvážit, zda je vhodné data ukládat do jedné, či více tabulek, zda mají být data v SQL přepisována, nebo přidávána, také množství dat, které bude zpracováváno a tedy i zatížení monitorovacího systému, SQL databáze a její velikost v dlouhodobém měřítku.

Jaká data je potřeba exportovat do SQL databáze:

- Veškerá nasbíraná data (pozor na obrovský objem dat)
- Jen některá data jako:
 - Teploty a tlaky systémů
 - Provozní hodiny zařízení
 - Hlavní hodnoty měřičů spotřeb

V jakém formátu mají být data do SQL zapisována a jak často:

- Jedna, nebo více tabulek, například pro každý modul zvlášť
- Struktura tabulek
 - Názvy a SQL typy sloupců
 - Primární indexy
 - o Obsah sloup ců vkládaný ch exportem, unikátnost identifikátorů a primárního indexu
- Perioda zápisu dat do SQL, přepisování dat v součinnosti s primárními indexy

🗾 Poznámka:

Návrh struktury SQL databáze bývá primárním problémem při zprovozňování výměny dat mezi různými informačními systémy. Na návrhu by se měly účastnit společně všechny dotčené strany, které budou data zpracovávat. Při nevhodném návrhu databáze může v dlouhodobém měřítku docházet k velké zátěži SQL databáze, k pomalému přístupu k datům a může být nasbírán obrovský objem špatně zpracovatelných dat.

4.1 Konzola správy ProSQL

Konzola správy ProSQL projektu

Samotná realizace exportu je v monitorovacím systému ProCop Designer 3.4 - Projekt realizována službou ProSQL, která je součástí monitorovacího systému ProCop od verze 3.4.

Správa projektu ProSQL je součástí návrhové části Designer monitorovacího systému ProCop. Otevřeme ji volbou Zobrazit \Správa ProSQL, nebo kombinací Ctrl+Alt+V v režimu designer. Prostředí konzoly je obdobou základního monitorovacího projektu datového serveru, stejně jako jeho obsluha.

Počítač, na kterém služba ProSQL běží (téměř vždy stejný, jako monitorovací, nejedná se o počítač SQL serveru) a projektu exportu je nastavitelné v designeru.



Spuštění správy ProSQL

Projekt ProSQL

Projektem ProSQL je soubor, obvykle s příponou ".ppq", ve kterém jsou uloženy veškeré potřebné informace k připojení k monitorovacímu systému (datovému serveru ProDat), k SQL databázi, včetně definice struktur tabulek a jejich exportovaných dat. Soubor je možné uložit kdekoliv, nejlépe však do adresáře monitorovacího projektu. Implicitně je projekt ProSQL umístěn v adresáři projektu.

ProCop Designer 3.4 - Projekt				
Soubor Projekt Zobrazit Uživatelé Okno Nápověda	🗋 🗋 🔦 < 🗔 🗐 🔍 🕨 🗖	🍠 🗲 🔍 🧶 🐘 🌆	🕼 🕼 📷 Moje	• •
⊎Vítejte W ProCop Designer Plugin version 3.4 WE:_	App\ProSQLProject\ProSQL.ppq	× 8	🔍 Vlastnosti a dynamiza	ace 🕂 🛛
Úgravy Hedání 🛛 🐟 🚕 🐰 🛽) 🖹 🕺 📓 💠 🦚 🧐 🚽 🔲	🖃 🦛 🦛 🕨 🗆 🝠 🦻		Hodnota
🖌 📲 Tento počítač	Název Typ	· · · · · · · · · · · · · · · · · · ·	Název	ProSQL Projekt
ProSQL Designer			Počítač	PETER-W7
🥬 Spuštěné úlohy	F TestProSQL MSSQL (Integrova	ané zabezpeče 🗷		
ProSQL Projekt	-			
▲ ·· E TestProSQL				
AutoExport				
▷ NormalExport				
▷ III ReportExport				

Správa ProSQL s otevřeným projektem a zobrazenými vlastnostmi připojení k datovému serveru

Práce s projektem v ProSQL je obdobná, jako s monitorovacím projektem. V levé části okna je zobrazen strom projektu, v pravé tabulkové zobrazení obsahu větve. Projekt lze rozčlenit na více částí, kdyby bylo potřeba data získávat z více datový serverů. Obvykle však stačí vložit jediný projekt a nastavit jeho parametry, viz následující podkapitola. Vlastnosti dané větve či řádku je možné zobrazit pomoci kláves a ctrl+Enter, v nabídce pak volbou Zobrazit/ Vlastnosti.

Parametry připojení k datovému serveru

Pro připojení k datovému serveru je potřeba zadat ve vlastnostech pouze jméno počítače, na kterém běží datový server monitorovacího systému ProDat. Pro lokální počítač ponecháme jméno prázdné. Oprávnění k přístupu je řešeno na úrovní DCOM práv operačního systému Windows.

Dále je možno zadat jméno projektu, které bude zobrazeno ve stromu ProSQL.

Parametry připojení k SQL databázi

V projektu je možno přidat SQL databáze, do kterých budou data exportována například klávesou Insert. Zrušit připojení je možné např. klávesou Delete. Obě operace jsou dostupné i v nástrojové liště.

Parametry připojení připojení k SQL serveru lze definovat ve vlastnostech (a Ctrl+Enter).

Nastavení parametry připojení k SQL databázi:

- Název pojmenování databáze, implicitně shodné se jménem databáze
- **Typ** výběr typu databáze, Microsoft SQL, MySQL, typ souboru a podobně
- Řetězec připojení konfigurace připojení databáze, je možné jej vytvořit pomocí systémového dialogu, který se otevře tlačítkem ... u řádku
- Primární klíče definuje, zda databáze podoporuje primární klíče (indexaci), soubory indexaci obvykle nepodporují, předvyplní se při přepnutí typu databáze
- Aktualizace hodnot definuje, zda databáze podoporuje aktualizaci (přepisování) hodnot, soubory aktualizaci obyykle penodporují před



Vlastnosti databáze - připojení k MS SQL databázi

- soubory aktualizaci obvykle nepodporují, předvyplní se při přepnutí typu databáze
- Adresa serveru IP adresa, nebo doménové jméno SQL serveru, instance definuje a sdělí správce SQL databáze
- Jméno databáze jméno SQL databáze vytvoří a sdělí správce SQL databáze
- Jméno uživatele uživatel SQL databáze s oprávněním pro čtení i zápis definuje a sdělí správce SQL databáze
- Heslo uživatele heslo uživatele SQL databáze definuje a sdělí správce SQL databáze

🗾 Poznámka:

Parametry se mohou dle zvoleného typu databáze lišit, jednotlivé parametry je možné zadat i v řetězci připojení, který lze upravit pomocí systémového dialogu.

4.2 Definice tabulek

Definice exportních tabulek

V projektu ProSQL je možné nadefinovat strukturu SQL databáze, jednotlivých tabulek a jejich sloupců. Tyto tabulky jsou zároveň i naplánovanými úlohami, které se dle svých vlastností spouštějí a zapisují data dle potřeby do SQL databáze.

Vytvoření tabulky - úlohy

Ve stromu v definované databázi je možné vytvořit jednotlivé tabulky se strukturou a vkládanými daty - řádky. Tato tabulka je zároveň naplánovaná úloha, která dle vlastností umožňuje automaticky exportovat data do databáze. Spuštěné exportní úlohy a jejich průběh se zobrazuje ve větvi Spuštěné úlohy, je-li projekt spuštěn a naplánovaná úloha běží.

Přidání tabulky možno provést například klávesou **Insert**, zrušit tabulku např. klávesou **Delete**. Obě operace jsou dostupné i v nástrojové liště. Zobrazí se výběr typu tabulky.

Typy exportních tabulek - úloh:

- Automatická zadáním indexu modulu budou exportována automaticky všechna dostupná aktuální data daného modulu projektu monitorovacího systému (vždy jedna proměnná na řádek), dle definice sloupců tabulky v naplánovaných časech
- **Normal** lze vytvořit kompletní seznam proměnných modulu, který je možné dále upravovat a promazat, vybraná aktuální data daného modulu projektu monitorovacího systému (vždy jedna proměnná na řádek), dle definice sloupců tabulky v naplánovaných časech
- **Report** uživatelsky ručně definovaná tabulka s různými proměnnými v řádku, vhodná například pro export dat jednoho modulu v jediném řádku (měřiče spotřeb, provozní hodiny, teploty a tlaky, atp.)

Vlastnosti tabulky - plánování úlohy

Tabulce lze měnit vlastnosti (a <u>Ctrl+Enter</u>) exportu do SQL databáze a naplánování.

Vlastnosti tabulky:

- Jméno pevné jméno tabulky v SQL databázi
- **Prefix modulu** prefix modulu monitorovacího projektu (pouze u typu Auto) definuje modul, jehož data budou kompletně automaticky exportována
- Přepisovat definuje, zda budou již dříve zapsaná data, která byla nalezena dle primárního indexu, opakovaně přepisována v SQL databázi - obraz modulu, tabulka se nebude v čase zvětšovat
- Pouze změny řádek se zapisuje pouze při změně hlavní hodnoty proměnné, nebo její validitě
 pouze u tabulek typu Auto a Normal

🚻 Vlastnosti a dynamizace	
	Hodnota
🗉 Name	ALFA_Report
Přepisovat	✓ <ano></ano>
Tolerance	1 m
Počáteční čas vůči aktuál…	*.* * * * *
🖩 Koncový čas vůči počáte	*.*.*.*
🗄 Tabulka sloupců	
🗉 Spustit úlohu nyní	Run !
🗆 Plánování	
🔲 Spouštět automaticky	Ano>
Typ spuštění	Perioda 🔳
🔟 Čas/Perioda spuštění	0:01:00
💷 Den v týdnu	Každý den 🔳

Vlastnosti tabulky - plánování úlohy

- **Počáteční čas vůči aktuálnímu** jen u tabulky typu Report, podrobněji bude rozebráno dále u tabulky sloupců
- Koncový čas vůči počátečnímu jen u tabulky typu Report, podrobněji bude rozebráno dále u tabulky sloupců
- Tabulka sloupců definuje strukturu tabulky v SQL databázi podrobněji bude rozebráno dále

Plánování úlohy:

- Spustit úlohu nyní umožní uživatelské okamžité spuštění úlohy pro vývoj a testování
- Spouštět úlohu automaticky povoluje automatické spouštění dle plánu
- **Typ spuštění** definuje, zda je úloha spuštěna jen v definovaný čas, nebo pravidelně s níže definovanou periodou
- Čas/Perioda spuštění definuje, nebo jak často, má být úloha automaticky spouštěna
- Den v týdnu definuje, které dny v týdnu má být úloha automaticky spouštěna

4.2.1 Definice struktury tabulky

Definice struktury tabulky

Tabulka v SQL serveru musí mít definovanou pevnou datovou strukturu, tedy jednotlivé sloupce, jejich názvy a typy, primární index. Dále je nutné definovat, jaká data budou z monitorovacího systému exportována do jakých sloupců SQL tabulky. Strukturu a typy dat definujeme ve vlastnostech tabulky v tabulce sloupců.

Vlastnosti tabulky

Ve vlastnostech tabulky (a <u>Ctrl+Enter</u>) je nutné nejprve nastavit strukturu tabulky v SQL databázi. Dle této struktury budou data exportována do SQL databáze.

Å Důležité upozornění!

Pokud je provedena významnější změna ve struktuře tabulky a tabulka je již v SQL databázi vytvořena, dojde k jejímu zrušení včetně exportovaných dat! Pro restrukturalizaci existující tabulky je nutné použít nástroje SQL serveru, aby bylo zabráněno ztrátě dat! Druhou možností je změnit jméno tabulky, data s novou strukturou budou ukládána do nové tabulky.

Tabulka sloupců

V tabulce sloupců reprezentuje jeden řádek právě jeden sloupec v definici řádků tabulky. Ne všechny definované sloupce jsou exportovány do SQL databáze, záleží to na jejich typu. Vytváření, kopírování a mazání sloupců (tedy řádků) lze provádět standardně v nástrojové liště, lokální nabídce, nebo klávesami **Insert** a **Delete**. Úpravy je možné provádět

myší, přímým psaním, nebo zahájit úpravy klávesou F2.

📲 Tal	bulka sloupců	1							x
	X 🗈 🖻	+ +)	< <i>m <u>m</u> m</i>						
Sloup	. Aktivní	Primární klíč	Název sloupce	Typ hodnoty		SQL Typ		Předvolba	
1	🔽 <ano></ano>	<ano></ano>	Name	Name	•	char(64)	•		
2	<ano></ano>	Ne>	Prefix	Prefix	•	text	•		
3	✓ <ano></ano>	Ne>	Value	Value	•	float	•	TeplVenkovni	
4	✓ <ano></ano>	Ne>	Volume	Value	•	float	•	TeplVenkovniGeometricka	=
5	✓ <ano></ano>	Ne>	OverValid	OverallValid	•	bit	Ŧ		
6	<ano></ano>	Ne>	Valid	Valid	•	bit	-	\$Volume	
7	<ano></ano>	Ne>	UpdateTime	UpdateTime	•	datetime	•	LastUpdate	
8	<ano></ano>	Ne>	MujUpdateTime	Value	•	datetime	•	LastUpdate	
9	<ano></ano>	<ano></ano>	WriteTime	WriteTime	•	datetime	•		
10	Ne>	<ne></ne>	Unknown	Unknown	•	none	•		
11	<ano></ano>	Ne>	TaskTime	TaskTime	•	datetime	•		
12	<ano></ano>	Ne>	Trend	Trend	•	text	•	\$Value	
13	<ano></ano>	Ne>	TrendValid	TrendValid	•	bit	•	\$Value	
14	Ano>	Ne>	Consumption	Consumption	•	text	•	\$Value	-
						l.			vřít

Definice sloupců tabulky

Změnu pořadí řádků (sloupců budoucí SQL tabulky) lze provádět pouze změnou pořadového čísla sloupce. Tabulka se okamžitě znovu uspořádá podle zadaného pořadí.

Každý řádek v tabulce (SQL sloupec) má následující vlastnosti:

- Sloupec pořadové číslo sloupce slouží jen pro přesun v tabulce sloupců, který se provede ihned po zadání číselné hodnoty pořadí
- Aktivní definuje, zda je sloupec exportován do SQL databáze pokud není aktivní, nebude dále zpracováván a kontrolován
- **Primární index** definuje, zda daný sloupec je součástí primárního indexu SQL tabulky, každá SQL tabulka musí mít alespoň jeden sloupec (nebo kombinaci sloupců) unikátní pro jednoznažné vyhledání, priorita sloupců při složeném primárním indexu je definována pořadím
- Unikátní zapíná kontrolu definice sloupce, zda musí být unikátní, kontrola se provádí pouze při překladu projektu
- Název sloupce jméno sloupce v SQL databázi (nesmí obvykle obsahovat jiné, než alfanumerické znaky, bez mezer, dle implementace SQL databáze)
- Typ hodnoty typ hodnoty, která bude z monitorovacího systému ProCop exportována
- **SQL Typ** SQL datový typ sloupce data z monitorovacího systému budou konvertována a uložena v tomto typu do tabulky
- Předvolba pokud v definici řádků nebude zadána žádná hodnota, bude použita tato předvolená

4.2.2 Typy hodnot pro export

Typy hodnot pro export

Pro definici sloupce je potřeba zadat způsob, jakým bude zadání buňky v řádku interpretováno při exportu do SQL. Tento způsob je potřeba vybrat ve sloupci **Typ hodnot** u každého sloupce při definici tabulky sloupců.

Typy hodnot pro export z monitorovacího systému ProCop

Jednotlivé sloupce v SQL databázi musí mít nastaven typ hodnoty, který je exportován z monitorovacího systému.

Typy hodnot exportovaných z monitorovacího systému:

- Unknown neznámý, není definován, sloupec se neexportuje do SQL databáze
 - Value hlavní hodnota v řádku zadané proměnné
 - Valid validita v řádku zadané proměnné znamená, že proměnná byla občerstvena v intervalu zadaném v monitorovacím projektu
 - **OverallValid** logický součin všech validit proměnných, zadaných v daném řádku (je li kterákoliv z hodnot v řádku nevalidní, bude nevalidní)

- UpdateTime příznak poslední obnovy hodnoty proměnné v mon. systému (.LastUpdate)
- TaskTime čas spuštění exportní úlohy
- Write Time čas zápisu řádku do SQL databáze
- Prefix prefix všech použitých proměnných v řádku (sloupec není exportován do SQL databáze, slouží jen aby nebylo nutné definovat stejné prefixy u všech sloupců v daném řádku, bývá zobrazen jako znak
 (a)
- Trend hodnota z trendu proměnné v čase definovaném v parametru úlohy
- TrendValid validita z trendu proměnné v čase definovaném v parametru úlohy
- Consumption rozdíl hodnot trendů v časech definovaném v parametrech úlohy
- ConsumptionValid validita hodnot trendů v časech definovaném v parametrech úlohy
- TrendBeginTime počáteční čas definovaný v parametru úlohy
- TrendEndTime koncový čas definovaný v parametru úlohy
- UserData uživatelská data libovolná data (text) zapsaná v řádku tabulky exportovaná do SQL databáze, nebere se z monitorovacího systému
- Time definovaná proměnná převedená na čas (datum se nepoužije)
- **Date** definovaná proměnná převedená na datum (čas je nulový)
- **DateOrNow** definovaná proměnná převedená na datum (čas je nulový), pokud je datum nedefinováno, zapíše se aktuální datum systému
- DateTime definovaná proměnná převedená na datum a čas
- Date Time Or Now definovaná proměnná převedená na datum a čas, pokud je datum a čas nedefinováno, zapíše se aktuální datum a čas systému
- Note poznámka libovolná data (text), zaznamenaný jen v projektu exportu, sloupec se neexportuje do SQL databáze

4.2.3 Datové typy sloupců v SQL

Každý sloupec tabulky v SQL databázi musí mít definován typ záznamu (datové reprezentace) hodnot. Tento typ je potřeba vybrat ve sloupci SQL Typ u každého sloupce při definici tabulky sloupců.

Datové typy sloupců v SQL databázi

Data v jednotlivých sloupcích získaná z monitorovacího systému dle sloupce "Typ hodnot", nebo definovaná v exportním projektu, jsou zaznamenávána v SQL databázi do sloupce zvoleného typu. Data jsou konvertována, je-li to možné do zvoleného SQL datového typu.

Typy sloupců v SQL databázi:

- none není definován, data se neexportují do SQL databáze
- **bit** logický typ, hodnoty 0 a 1 (false, true)
- tinyint 1 bajt, bez znaménka 0-255
- **smallint** 2 bajty, se znaménkem
- int 4 bajty se znaménkem
- **bigint** 8 bajtů se znaménkem
- float 4 nebo 8 bajtů dle velikosti, reálné číslo s plovoucí desetinnou čárkou
- real 4 bajty, reálné číslo s plovoucí desetinnou čárkou
- text text libovolné délky
- char(64) text maximálně 64 znaků dlouhý
- datetime datum a čas

🕭 Důležité upozornění!

Datové typy v SQL databázi se mohou lišit dle konkrétní implementace SQL serveru.

4.3 Report tabulka

Report Tabulka

V tabulka typu Report je především určena pro export dat z více modulů současně, kde modul má přiřazen jeden či

několik řádků a proměnné daného modulu jsou organizovány ve sloupcích. Z tohoto důvodu byl do definice sloupců tabulky doplněn typ sloupce "Prefix, který se neexportuje do SQL databáze, ale slouží jen jako prefix proměnných definovaných na daném řádku. Není tedy nutné vypisovat do jednotlivých buněk celé jméno proměnné, stačí zadat jméno proměnné bez prefix a ve sloupci prefix zadat konkrétní prefix řádku. Při exportu je prefix z daného řádku ke všem proměnným na řádku automaticky doplněn. Existují i další metody, jak si ušetřit práci s opisování stejných textů, ty budou popsány dále.

Názvy sloupců v definici řádků tabulky

Ve stromu exportního projektu se v tabulce typu Report nachází větev řádky. Ta obsahuje předpis všech řádků. které se pokusí systém exportovat při každém spuštění úlohy. V záhlaví řádků jsou zobrazeny sloupce tak, jak jsou definovány v tabulce sloupců dané SQL tabulky. Jejich názvy jsou však v některých případech označeny znakem 🔛 a někdy různě uvozeny.

▲ ··· ▲ Tento počítač ▲ ··· ★ ProSQL Designer	*Name	/Prefix=@/	Value	Volume		<overvalid></overvalid>	Valid
■	y ↔ IALB1 ↔ \$IALB\$ Report ↔ IALB	IALB IALB \$IALB\$	<@.TeplVenkovni> \$@.TeplVenkovniG \$	> <@.TeplV eo <@.TeplV <@.TeplV	enkovniGeo enkovniGeo enkovniGeo	<> < <	\$@.TeplVenkovni\$ <\$@.TeplVenkovniGe <\$@.TeplVenkovniGe
I	Rádky						
L	Hádky UpdateTime	MujUpdateTime	<*WriteTime>		<tasktime></tasktime>	Trend	TrendValid

Ukázka definice řádků tabulky typu Report s využitím předvoleb a odkazů

Doplňky k názvu sloupce mají přispět ke snadnější orientaci v tabulce a jsou tyto:

- **'*'** označuje sloupce, které jsou součástí primárního indexu, priorita sloupců je určena pořadím definice v tabulce sloupců
- / Název sloupce/ takto označený sloupec nebude exportován do SQL databáze, slouží jen k účelům exportu (prefix, poznámka, neznámý typ hodnot)
- / Název sloupce=@/ Název sloupce=@ poukazuje na to, že sloupec je typu prefix a zobrazení znaku '@' v buňkách znamená, že zde bude automaticky při exportu doplněn obsah sloupce typu Prefix v daném řádku
- **<Název sloupce>** uvození jména sloupce do špičatých závorek poukazuje na to, že hodnoty jsou vždy předvolené a do sloupce se při definic řádků nic nezadává (sloupce typu WriteTime čas zápisu řádku do SQL, TaskTime čas spuštění úlohy, atp.)

Vkládání, kopírování a mazání definic řádků

Jednotlivé řádky lze do tabulky přidávat Insert, v tabulce posouvat Alt+Up Alt+Down, kopírovat Ctrl+C Ctrl+V i mazat Delete. Operace jsou rovněž dostupné přes hlavní i lokální nabídku a nástrojovou lištu. Nechtěně provedené operace je možné vrátit zpět Ctrl+Z.

Jednotlivé sloupce v řádku je potřeba vyplnit požadovanou hodnotou dle typu sloupce, která bude sloužit k exportu dat do SQL databáze. Pokud je v definici sloupců zadán i sloupec typu **Prefix**, není potřeba ve sloupcích, které vyžadují zadání proměnné, prefix zadávat.

Předvolené hodnoty v definici řádků

V definici sloupců tabulky je možné zadat ve sloupci **Předvolba** libovolný text. Pokud definujeme nový řádek a ponecháme některé sloupce nevyplněné, použije se tato předvolba pro daný sloupec a zobrazí se ve špičatých závorkách, např. <<u>@.TeplVenkovni></u>. Takto zobrazená hodnota v buňce říká, že daný sloupec nebyl zadán a byla použita předvolba z definice sloupců tabulky a znak '@' nahrazuje text ve sloupci typu Prefix daného řádku.

🚖 Tip!

Pokud budou exportována data z více modulů stejného typu, je vhodné definovat sloupec typu prefix, do kterého se budou uvádět jen prefixy konkrétních modulů pro daný řádek. Sloupce pak mohou být definovány jako jednotlivé proměnné modulu, jejichž jméno bude u jednotlivých modulů stejného typu pravděpodobně shodné. Proto je vhodné jméno proměnné zadat jako předvolbu při definici sloupců. Při vkládání definic řádků do tabulky tak bude potřeba zadat pouze prefix modulu, ostatní sloupce se doplní přednastavenou hodnotou.

Odkazy mezi sloupci při definici řádků

Při definici řádků může nastat potřeba, aby se dva či více sloupců odkazovaly na stejnou proměnnou. Typicky například při použití typu sloupců Value, Valid a UpdateTime, kde z jedné proměnné lze získat hodnotu, její validitu a čas, kdy byla naposledy občerstvena. Takto definované tři sloupce vyžadují zadání u každého řádku totožně třikrát tutéž proměnnou.

Aby se předešlo zbytečným duplicitám při zadávání, byl zaveden odkaz na hodnotu jiného sloupce ve stejném řádku. Do buňky stačí zadat text **\$Název sloupce** a při zadání či změně hodnoty v odkazovaném sloupci se změna automaticky promítne do sloupce s odkazem. Odkazované hodnoty jsou pak uvozeny znaky \$: **\$@.TeplVenkovni\$**.

Odkaz na hodnotu jiného sloupce stejného řádku lze použít i v předvolbě. Takováto hodnota bude pro přehlednost zobrazena takto: <\$@.TeplVenkovni\$>.

Zadání prázdné proměnné

Při kontrole (kompilaci) se kontroluje existence všech proměnných a zadání všech sloupců. Pokud není proměnná, nebo hodnota v buňce zadána, je vyhlášena chyba, aby nedošlo omylem k opomenutí zadání. Pokud však nemá být hodnota daného sloupce a řádku do SQL skutečně exportována (nemá být například použita předvolba), je potřeba zadat namísto prázdné buňky znak '\$', nebo text '\$EMPTY'. Takto vyplněná buňka nebude kontrolována a do SQL databáze nebude zapisována.



Odkazy na sloupce, lze provádět při definici řádků i v předvolbách při definici sloupců. Pokud je však potřeba do konkrétní buňky zadat jinou proměnnou, než je předvolená, stačí její jméno zadat do buňky. Prefix modulu se doplní automaticky ze sloupce typu Prefix, pokud je zadán.

4.4 Přiklad exportu měřičů spotřeb

Příklad exportu měřičů spotřeb

Typickým příkladem exportu dat do nadřízených informačních a fakturačních systémů přes SQL databázi jsou odečty měřičů spotřeb, jako jsou měřiče tepla, chladu, elektřiny, plynu, vodoměry a mnohé další. Následující příklad uvádí jednu z mnoha možností, jak poskytovat odečtené hodnoty z monitorovacího systému

do SQL databáze. Z této databáze může informační systém třetí strany získávat potřebná data.

🗾 Poznámka:

Konkrétní definice struktury databáze záleží na domluvě s administrátory informačního či fakturačního systému a je velice variabilní.

Popis exportu měřičů spotřeb

Měřiče spotřeb i různých výrobců a typů měří veličiny jako množství spotřebované energie, objemu média, okamžité veličiny, napětí, teploty a podobně. Pro náš příklad se zaměříme na měřiče tepla (chladu) a vodoměry. Zadefinujeme strukturu databáze, která sloučí rozdílné vlastnosti různých měřičů do jediné tabulky. Tato tabulka - úloha, která jí vyplňuje pak může být periodicky naplánována, například každý den v 6 hodin ráno.

Pro strukturu tabulky tak budeme potřebovat znát exportované veličiny, identifikátor měřiče spotřeb a datum exportovaných hodnot.

V novém projektu si nejprve vytvoříme novou databázi, zadáme jméno počítače SQL serveru, instanci a přihlašovací údaje do databáze. Vyzkoušíme funkčnost připojení v lokální nabídce ve stromu na větvi databáze. Zobrazí se jednoduchý prohlížeč SQL databáze, případně chybové hlášení s popisem chyby.

Struktura tabulky

Do databáze si přidáme definici tabulky typu Report, které zadáme jméno, zadefinujeme čas spuštění na 6 hodin každý den a otevřeme Tabulku sloupců.

📲 Tab	ulka sloupc	ů						
	% 💼 🖻	+ +	× n n n					
Sloup	Aktivní	Primární klíč	Název sloupce	Typ hodnoty		SQL Typ		Předvolba
1	📝 <ano></ano>	<ano></ano>	DateIndex	DateOrNow	Ŧ	datetime	•	LastUpdate
2	✓ <ano></ano>	Ne>	DateTime	DateTimeOrNow	•	datetime	-	\$DateIndex
3	✓ <ano></ano>	✓ <ano></ano>	Modul	Name	•	char(64)	•	\$Prefix
4	✓ <ano></ano>	Ne>	Prefix	Prefix	•	text	•	
5	✓ <ano></ano>	Ne>	ID	Value	•	text	•	CustomerNumber
6	✓ <ano></ano>	<ne></ne>	CustomerNumber	UserData	•	text	•	
7	✓ <ano></ano>	Ne>	Energy	Value	•	real	•	Energy
8	✓ <ano></ano>	Ne>	Volume	Value	•	real	-	Volume
9	✓ <ano></ano>	<ne></ne>	Power	Value	•	real	•	Power
10	✓ <ano></ano>	<ne></ne>	Flow	Value	Ŧ	real	-	Flow
11	✓ <ano></ano>	Ne>	OutputTemperature	Value	Ŧ	real	•	OutputTemperature
12	<ano></ano>	Ne>	ReturnTemperature	Value	Ŧ	real	•	ReturnTemperature
13	✓ <ano></ano>	<ne></ne>	Valid	OverallValid	Ŧ	bit	•	
14	✓ <ano></ano>	<ne></ne>	Note	Note	Ŧ	text	•	
	•	•	,					✓ Zavřít

Ukázka definice struktury SQL tabulky, jednotlivých sloupců

Vytvoříme jednotlivé sloupce tabulky (zobrazují se jako řádky) a zadefinujeme jim parametry dle následujícího obrázku.

Strukturu tabulky a význam sloupců se pokusíme rozebrat důkladněji, sloupec po sloupci:

- **Date Index** sloupec je součástí primárního indexu, do kterého se zapíše datum, zadané proměnné, předvolba je LastUpdate, tedy poslední obnova proměnné (modulu)
- DateTime datum a čas poslední obnova proměnné (modulu), předvolba je definována jako odkaz na hodnotu ve sloupci DateIndex
- **Modul** libovolný text jednoznačně identifikující konkrétní měřič tepla je součástí primárního indexu předvolba je reference na sloupec prefix
- Prefix sloupec, který nebude v SQL databázi a uvede se do něj prefix příslušného modulu pro ostatní proměnné v řádku, bez předvolby, musí se vždy zadat
- ID hodnota z měřiče sériové číslo předvolba proměnná CustomerNumber
- CustomerNumber hodnota zadaná ručně při exportu kontrolní zákaznické číslo měřícího místa
- Energy hodnota z měřiče kumulovaná energie předvolba: proměnná Energy
- Volume hodnota z měřiče kumulovaný objem předvolba: proměnná Volume
- Power hodnota z měřiče okamžitý výkon předvolba: proměnná Power
- Flow hodnota z měřiče okamžitý průtok předvolba: proměnná Flow
- OutputTemperature hodnota z měřiče teplota přívod- předvolba: proměnná OutputTemperature
- ReturnTemperature hodnota z měřiče teplota zpátečka předvolba: proměnná ReturnTemperature
- OverAllValid validita všech hodnot z měřiče vypočtená hodnota ze všech validit hodnot proměnných na řádku
- Note pouze poznámka u každého řádku v definici řádků, neexportuje se do SQL databáze

K primárnímu indexu uveďme, že se skládá ze dvou sloupců - DateIndex a Modul. Při spuštění úlohy se projdou všechny řádky, vyhodnotí se sloupce dle definice typů a zapíší se do SQL databáze. Takto navržený primární index nám zaručí, že pro každý měřič se k danému datu zapíše jen jediný řádek. Pokud jsou již data k danému datu pro daný modul v databázi zaznamenána, záleží pak na příznaku "Přepisovat" v parametrech úlohy, zda se řádek přepíše novějšími daty, či nikoliv. V případě odečtů měřičů není důvod při vícenásobném spuštění úlohy ve stejném dni data přepisovat.

Definice řádků tabulky - měřičů

Máme-li definovánu strukturu databáze, můžeme přejít k definici řádků - měřičů. Každý řádek nám bude reprezentovat jedem měřič, sloupce pak hodnoty z konkrétního měřiče. Následující obrázek ukazuje zadání řádků - měřičů.

Export	do	SQL	data	báze
--------	----	-----	------	------

DateIndex		DateTir	ne	*Modul		/Prefix=@/	ID	Cust	omerNumbe	r Energy	
⇔ <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$DK10M1	124\$>	DK10M124	<@.CustomerNumber>	9981	24	<@.Ene	ergy>
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$IOS967	M024\$>	IOS967M024	<@.CustomerNumber>	9988	324	<@.Ene	ergy:
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$OS616M	1068\$>	OS616M068	@.MeasureId	9686	5	<@.Ene	ergy:
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	OS616M06	8A	OS616M068	<@.CustomerNumber>	6825	5100	\$	
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	OS616M06	8B	OS616M068	<@.CustomerNumber>	7982	395	\$	
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$DK5BV1	10\$>	DK5BV110	<@.CustomerNumber>	9960	327	\$	
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$IOS967	M002\$>	IOS967M002	<@.CustomerNumber>	9988	321	<@.Ene	ergy
< <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$IOS967	M031\$>	IOS967M031	<@.CustomerNumber>	9986	31	<@.Ene	ergy
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$IOS967	M027\$>	IOS967M027	<@.CustomerNumber>	9986	27	<@.Ene	ergy
\leftrightarrow <@.LastUp	date>	<\$@.L	astUpdate\$>	<\$IOS967	M076\$>	IOS967M076	<@.CustomerNumber>	9986	26	<@.Ene	ergy
			-	La			L				
	Volume	2	Power	Flow	OutputTe	emperature	ReturnTemperature		<valid></valid>	/Note/	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outp	outTemperature	<@.ReturnTemperat	ure>	<>	1.	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outp	outTemperature	emperature> <@.ReturnTemperatu		$\langle \rangle$	2.	
	<@.Vo	olume>	<@.Power>	<@.Flow>	@.InputT	Temperature	@.OutputTemperatur	re	$\langle \rangle$	3.	
	@.A		\$	\$	\$	\$			<>	4.	
41.	@.B		\$	\$	\$		\$		<>	5.	
	@.Inpu	uts	\$	\$	\$		\$		<>	6.	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outp	outTemperature:	< @.ReturnTemperat	ure>	<>	7.	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outp	outTemperature	< @.ReturnTemperat	ure>	<>	<>	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outp	outTemperature:	< @.ReturnTemperat	ure>	<>	<>	
	<@.Vo	olume>	<@.Power>	<@.Flow>	<@.Outr	outTemperature)	< @.ReturnTemperat	ure>	$\langle \rangle$	\diamond	

Ukázka definice exportu měřičů do SOL databáze, definice jednotlivých řádků - měřičů

Ve sloupci poznámka / Note/ jsou záměrně zapsány čísla řádků pro snazší popis. V tomto sloupci může být uvedeno cokoliv, dále se nezpracovává a do SQL databáze se neexportuje.

V prvním a druhém řádku pro měřič DK10M124 a IOS967M04 stačilo zadat do sloupce prefix uvedené prefixy modulů a kontrolní zákaznické číslo. Jak je ze špičatých závorek a referencí (\$) patrné, ostatní sloupce nebylo potřeba vyplnit, byly použity předvolby.

Na třetím řádku je uveden měřič tepla, který má jiný název proměnné pro výrobní číslo. Proto bylo nutné zadat ještě MeasureId do sloupce **ID**. Dále má rozdílné názvy teplot, proto i tyto bylo potřeba zadat ručně. Ostatní hodnoty jsou nevyplněné (z předvoleb).

Na čtvrtém a pátém řádku jsou ve skutečnosti údaje z měřiče, který je uveden na třetím řádku. Na jeho dva impulsní vstupy A a B jsou připojeny vodoměry. Tyto jsou však měřidla s vlastní identifikací, proto musí být na různých řádcích. Měří pouze objem. Ostatní veličiny nejsou zadány. Při kontrole by však vznikla chyba, že proměnná nebyla zadána. Proto na místo prázdných proměnných musíme uvést znak '\$', nebo text '\$EMPTY'. Všechny tři řádky mají shodný prefix, ale modul musí být unikátní z důvodu primárního indexu. Proto do sloupce Modul je v tomto případě zapsán název modulu s rozšířením a písmeno impulsního vstupu.

Na šestém řádku je umístěn nezávislý měřič s imuplsním vstupem a připojeným vodoměrem. Hodnota objemu je v proměnné Inputs, nikoliv Volume, proto je nutné jméno proměnné explicitně do buňky zadat. Ostatní proměnné mají zůstat prázdné, proto je použit znak '\$'.

Od sedmého řádku jsou zase uvedeny standardní měřiče tepla, kde není potřeba zadat nic jiného, než prefix modulu a zákaznické číslo měřiče, jako u prvních dvou.

Spuštění a kontrola výsledku

Naplánovanou úlohu spustíme. Ta otevře databázi a začne dle definice procházet každý řádek - měřič, jeden po druhém. Hodnoty proměnných získá z monitorovacího systému ProCop dle jména v buňce, předvoleného hodnoty či odkazu a prefixu v daném řádku. Pokusí se dle primárního indexu nalézt řádek v databázi se stejným datem a názvem modulu. Pokud jej nenajde, celý řádek do databáze přidá, pokud ano, dle příznaku "Přepisovat" zadaném ve vlastnostech úlohy řádek přepíše, nebo vynechá a jde zpracovávat další řádek v definici. Takto postupuje, až projde všechny definované řádky - měřiče.

Průběh exportu můžeme sledovat ve stromu projektu ve větvi "Spuštěné úlohy". Až je úloha dokončena, je možné zkontrolovat data v SQL databázi, například jednoduchým prohlížečem, který lze otevřít volbou z lokální nabídky databáze ve stromu projektu.

Å Důležité upozornění!

Případné chyby vzniklé při běhu úlohy se zapisují do systémového zápisníku Windows, do sekce AlfaLog pod názvem

aplikace ProSQL.
5 Provoz v doméně bez lokálního uživ.

Komponenty systému ProCop jsou standardně provozován pod identitou lokálního uživatele ProCop, který je obvykle lokálním administrátorem počítače. Oprávnění tohoto uživatele mohou být snížena až na úroveň skupiny Users (popis konfiguračních kroků není součástí tohoto dokumentu). V případě restriktivních doménových politik je v některých případech požadován provoz bez tohoto uživatele. Změnu konfigurace je nutno v tomto případě provést ručně.

Podmínky pro úspěšnou konfiguraci

- pro změnu konfigurací jsou nutná administrátorská oprávnění k počítačům
- je provedena instalace systému ProCop typu "Standardní monitorování" na počítači dále označovaném jako server
- volitelně instalace systému ProCop typu "Síťové pracoviště" na počítači označovaném dále jako klient
- na obou počítačích provoz pod stejným nebo různým účtem s oprávněními Domain User
- instalován systém ProCop verze 3.3.8.6 a novější na všech strojích

Podporované kombinace operačních systémů

Operační systémy	Operační systémy Varianta 1		Varianta 3			
Server	Windows XP Pro SP 3	Windows 7 Pro a novější	Windows 7 Pro a novější			
Klient	Windows XP Pro SP 3	Windows XP Pro SP 3	Windows 7 Pro a novější			

5.1 Úpravy serveru pro provoz bez uživatele ProCop

Změna indentity služeb systému ProCop

Identita služeb je standardně nastavena na lokálního uživatele ProCop. Pro změnu je nutno kliknout Start Menu/Tento počítač pravým tlačítkem myši volba Spravovat. V konzole Správa počítače v položce Služby a aplikace otevřete Služby. Na pravé straně v seznamu služeb najděte ProCop Access Server a volbou v menu Akce/Zastavit zastavíte všechny služby systému ProCop. Volbou Vlastnosti pravým tlačítkem myši na kartě Přihlášení změníte učet pro přihlášení na Místní systémový účet.

Nastaveni identity služeb:
i ProCop Access Server version 3.3 (Local Co
Přihlášení Zotavení Závislosti
přihlášení:
ní systémový účet

2

Obecné Přihlášení Zotavení Z	ávislosti
Účet pro přihlášení:	
Místní systémový účet Povolit službě používání plo	shy
◯ <u>I</u> ento účet:	P <u>r</u> ocházet
Heslo:	
Pot <u>v</u> rzení hesla:	
Můžete povolit nebo zakázat spouš hardwarových profilech: Hardwarový profil	tění této služby v následujících Služba
<u>M</u> ůžete povolit nebo zakázat spouš hardwarových profilech: Hardwarový profil Profil 1	tění této služby v následujících Služba Povoleno
<u>M</u> ůžete povolit nebo zakázat spouš hardwarových profilech: Hardwarový profil Profil 1	tění této služby v následujících Služba Povoleno <u>P</u> ovolit <u>Z</u> akázat

Nastavení je nutné provést pro všechny služby systému ProCop. Po konfiguraci je možné spustit služby obvykle spuštěním služby ProCop Data Server, která díky vazbám spustí i ostatní služby systému ProCop.

Úpravy zástupců Panelu nástrojů ProCop

Vlastnost

Na ploše systému Windows je standardně nainstalován zástupce pro spouštění Panelu nástrojů. Kliknutím pravým tlačítkem myši na zástupci volbou Vlastnosti se otevře dialog pro konfiguraci tohoto zástupce. Změníme položku Cíl na novou hodnotu (i s uvozovkami) a potvrdíme tlačítkem OK:

"C:\ProCop 3.3\ProCop.exe" /connect

Obdobně v nabídce Start podnabídka Po spuštění položka Panel nástrojů ProCop stiskem pravého tlačítka myši otevřeme lokální nabídku. Volbou Vlastnosti upravíme cíl odkazu stejně jako v předchozím případě.

Posledním je zástupce v nabídce Start je podnabídka Monitorovací systém ProCop položka Panel nástrojů ProCop. Cíl změníme stejně jako v předchozích případech. Odzkoušíme funkčnost všech odkazů.

Úprava spuštění ohlašování stavu projektu ProTray

Spustím editor registru regedit.exe například zadáním tohoto příkazu do příkazové řádky. Najdeme klíč:

HKEY LOCAL MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Na pravé straně najdete textovou hodnotu ProCop Tray. Dvojitým klikem myši otevřeme editor a nastavíme hodnotu na (bez uvozovek):

C:\ProCop 3.3\ProTray.exe

Tím jsou všechny úpravy pro lokální monitorování hotovy. Po restartu počítače by mělo všechno fungovat správně. Následně je možno odstranit lokálního uživatele ProCop ze systému Windows.

Konfigurace DCOM pro provoz síťového pracoviště

V případě požadavku na provoz se síťovým pracovištěm je nutno nakonfigurovat přístupová oprávnění pro přístup službami DCOM na tento server. Spustíme ovládací konzolu dcomcnfg.exe například zadáním příkazu do příkazové řádky Windows. Najdeme ve stromu Službu komponent větev Počítače položka Tento počítač. Pravým tlačítkem otevřeme nabídku a volbou Vlastnosti otevřeme dialogové okno. V záložce zabezpečení COM jsou k dispozici tlačítka Upravit omezení a Upravit výchozí. Provedeme konfiguraci shodně v dialozích pod oběma tlačítky:

- stiskem tlačítka Přidat přidáme skupinu uživatelů Domain Users
- následně vybereme přidanou položku v prvém seznamu
- ve sloupci Povolit ve druhém seznamu zatrhneme všechny dostupné volby (Místní i vzdálená aktivace i spouštění)
- po restartu je server připraven na připojení síťových pracovišť



					-			
	Tento počítač						?	×
	Obecné	1	Možnos	ti	Výc	hozí vlastno	sti	l
	Výchozí prot	okoly	MS	DTC	Za	bezpečení C	ОM	
	Přístupová op	rávnění –			×/			
	Můzete upr aplikace rov oprávnění.	avit, komu /něž může	i je povolei ete nastavi	n výchozi t omezení,	přistup k , která urč	aplikacim. P sují jejich vlas	ro stní	
			Upravit o	mezení	Up	ravit výchoz	í	
	Spouštěcí a a Můžete upr nebo aktiva omezení, kl	ktivační o avit, komu ice objekt erá určují	oprávnění i je standai ů. Pro aplik jejich vlas	dně povol kace rovně tní oprávn	eno spou éž můžete iění.	štění aplikac nastavit	í	
		(Upravit o	mezení	Up	ravit výchoz	í	
právnění l	ke spouštění				2 🗙			
Omezení zał	hezpečení			l				
unezeni zai					1			
Nazev skuj	introtoro (TECT VEVA	atele: desisistrati			-			
Admin C 😢 Doma	in Users (ALFA\Doma	ain Users)	18	>				
🕵 E very	one) F	oužít	
		<u> </u>	lat	0 <u>d</u> ebra	it			
Op <u>r</u> ávnění	pro Domain Users		Povolit	Odepřít	_			
Místní s Vadáloni	puštění á opuštění							
Místní a	e spastern iktivace							
Vzdálen	á aktivace							
,								
			OK	Sto	rno			

Nastavení výchozí pro spouštěcí a aktivační oprávnění:

Tento počítač			? X	
Obecné	Možnosti	Výchozí vla:	stnosti	
Výchozí protokoly	MSDTC	Zabezpeče	ní COM	
Přístupová oprávnění -				
Můžete upravit, kom aplikace rovněž můži oprávnění.	u je povolen výchozí ete nastavit omezení,	přístup k aplikací: , která určují jejich	m. Pro vlastní	
	Upravit omezení	Upravit výc	hozí	
Spouštěcí a aktivační Můžete upravit, kom nebo aktivace objek omezení, která určuj	oprávnění u je standardně povol ů. Pro aplikace rovně í jejich vlastní oprávn Upravit omezení…	eno spouštění apl šž můžete nastavit ění. Upravit výc	ikací hozí	
	Oprávnění ke s	pouštění		2
	Výchozí zabezpe	čení		
	Názov okupinu i	l nebo iméno uživati	ale:	
	Administrat	ore (TEST VP\Adr	pinistrators)	
	- Domain Us	ers (ALFA\Domain	n Users)	
		IVE		
	🕵 SYSTEM			
			<u>P</u> řidat	0 <u>d</u> ebrat
	Op <u>r</u> ávnění pro [Domain Users	Perolit	Odepřít
	Místní spuště Vzdálené spu Místní aktiva Vzdálená akt	éní ištění ice ivace		
			ОК	Storno

5.2 Úpravy klienta pro provoz bez uživatele ProCop

Úpravy zástupců Panelu nástrojů ProCop

Na ploše systému Windows je nainstalován zástupce pro spouštění Panelu nástrojů. Kliknutím pravým tlačítkem myši na zástupci volbou Vlastnosti se otevře dialog pro konfiguraci tohoto zástupce. Změníme položku Cíl na novou hodnotu a potvrdíme tlačítkem OK:

"C:\ProCop 3.3\ProCop.exe" /remote=NAZEV_POCITACE

kde NAZEV_POCITACE je doménové jméno počítače serveru.

Obdobně v nabídce Start podnabídka Po spuštění položka Panel nástrojů ProCop stiskem pravého tlačítka myši otevřeme lokální nabídku. Volbou Vlastnosti upravíme cíl odkazu stejně jako v předchozím případě.

Posledním je zástupce v nabídce Start podnabídka Monitorovací systém ProCop položka Panel nástrojů ProCop. Cíl změníme jako v předchozích případech. Odzkoušíme funkčnost všech odkazů.

6 Data v SQL

ProCop od verze 3.5 disponuje možností ukládat data nejen v BDE databázi, ale také s využitím Microsoft SQL Serveru.

Podporované SQL servery

V současnosti je možné použít SQL servery společnosti Microsoft a to verze 2008, 2012, 2014 a 2016. V případě využití volné edice Express je nutno počítat s omezením velikosti databáze a rychlostí. Pro udržování konstantní velikosti databáze SQL je možné použít archivaci trendů 53 do dalších databází.

Podporované SQL Servery:

- Microsoft SQL Server 2012/2014/2016 Standard (případně vyšší verze)
 - o limitováno pouze velikostí úložiště, velikostí paměti a počtem licencovaných jader
- Microsoft SQL Server 2012/2014/2016 Expres (zdarma)
 - o limit 10GB velikost databáze, 1GB RAM

Å Důležité upozornění!

Licence SQL databází nejsou součástí monitorovacího systému ProCop a řídí se licenčními podmínkami a pravidly výrobce. SQL databáze Standard je v současnosti licencována na uživatele, nebo na jádra procesoru. Licencování na uživatele nelze použít v případě publikování na webu (ProWeb), případně je neekonomické, jelikož se do uživatelů počítají veškeré zdroje dat (stanice, AlfaBoxy, importy) plus uživatelé.

対 Tip!

Pro zjištění velikosti databáze (trendů), postačí provést kompilaci celého projektu a na konci překladu modulů se od verze 3.6 zobrazuje statistika period trendů a velikosti dat v SQL za 1 rok.

Scénáře nasazení

Využití SQL databáze Monitorovacím systémem ProCop lze v různých variantách:

- MS SQL je nainstalována lokálně nebo na jiném serveru, je k dispozici oprávnění správce (sa)
 - ProCop je správcem serveru (sa), umožňuje vytvářet databáze s archivy trendů
 - o instalační program vytvoří databázi ProCopDB v SQL serveru a také ji připraví k použití
- MS SQL je nainstalována lokálně nebo na jiném serveru, není k dispozici oprávnění správce
 - nejednodušší je požádat správce SQL serveru o vytvoření databáze skriptem "Create ProCop Database.sql"
 - nebude možné vytvářet archivy trendů, ProCop nemá oprávnění vytvářet další databáze v SQL serveru
 - pokud to správce umožní, postačuje nastavit TRUSTWORTHY ON na celém SQL serveru a archivace bude možná
 - ALTER DATABASE ProCopDB SET TRUSTWORTHY ON
- Lokální Microsoft SQL Server 2012/2014/2014 Express instaluje monitorovací systém ProCop
 - o je nutné mít k dispozici instalační soubory (setup.exe) a ten odkázat při instalaci
 - o ProCop je správcem serveru (sa), umožňuje vytvářet databáze s archivy trendů
 - instalační program nainstaluje SQL Server, instanci ProCop, vytvoří databázi ProCopDB a tu také připraví k použití

6.1 Instalace SQL databáze

Instalační program umožňuje nastavit (nainstalovat) a připravit SQL databázi:

- Povolit nastavení přístupu do ProCop Cloudu
 - umožní volit v designeru na tomto počítači, zda exportovat data do cloudu (procloud.alfamik.cz), či nikoliv
 - pokud není tato volba zatržena, nebude možné projekt z tohoto počítače exportovat do cloudu, lze změnit pouze v registry
- Žádná akce s SQL databází
 - \circ nic nedělá, vše bude nutné provést z designeru, pokud bude potřeba nastavit SQL
- Konfigurovat SQL databázi pro systém ProCop
 - o nakonfiguruje již existující SQL server, je potřeba oprávnění správce (sa)
 - následně bude potřeba zadat:
 - SQL Server jméno (IP adresa) počítače, na kterém je nainstalován SQL server (localhost = tento počítač)
 - Jméno instance SQL serverů může být na jednom počítači instalováno více pak jsou pojmenovány, pokud je jen jeden (Default)
 - Heslo správce (sa) heslo nejvyššího správce celé instance SQL serveru, pokud jej nemáme musíme požádat správce, aby nám vytvořil databázi ručně
 - Název databáze volitelný, obvykle ProCopDB, nebo jakýkoliv jiný
 - Heslo uživatele ProCop při přípravě se vytváří uživatel ProCop, vlastník a kompletní správce databáze (nikoliv instance), můžeme ponechat prázdné, bude shodné s heslem správce (sa)
 - Instalovat MS SQL 2014 Express Edition (nutno mít Setup.exe)
 - o nainstaluje a nakonfiguruje bezobslužně MS SQL Express
 - následně bude potřeba zadat:
 - SQL Server jméno (IP adresa) počítače, na který se má nainstalován SQL server (obvykle localhost = tento počítač)
 - Jméno instance jméno této instalace SQL serveru, nejlépe ponecháme ProCop, nebo prázdné
 - Heslo správce (sa) heslo nejvyššího správce celé instance SQL serveru
 - Název databáze volitelný, obvykle ProCopDB, nebo jakýkoliv jiný
 - Heslo uživatele ProCop při přípravě se vytváří uživatel ProCop, vlastník a kompletní správce databáze (nikoliv instance), můžeme ponechat prázdné, bude shodné s heslem správce (sa)
 - Cestu k instalačním souborům (Setup.exe)
 - Adresář pro SQL Data zde budou umístěny databáze
 - Lze povolit zobrazení uživatelského rozhraní a ručně provést instalaci SQL serveru

ঢ় Moni	torovací systém Pro	oCop version 3.6.3.1	- • • ×
J	2 🔿	ProCloud a SQL Nastavte využívání ProCop Cloud a SQL	
	Povolit nastave	ení přístupu do ProCop Cloudu	
	Nastavte zda využ	ívat ukládání dat do SQL	
	Žádné akce s	s SQL databází	
	🔘 Konfigurovat	t SQL databázi pro systém ProCop	
	🔘 Instalovat M	S SQL 2014 Express Editition (nutno mít Setup.exe)	
Nullsoft I	nstall System v3.0b2		
		< <u>Z</u> pět Další >	Storno

Instalace ProCop - volba operace s SQL databází

🚖 Tip!

Instalaci a konfiguraci SQL serveru lze také provést dávkovými soubory sqlInstall.bat a sqlPrepare.bat v podadresáři Register po instalaci.

6.2 Nastavení projektu pro ukládání dat do SQL

ProCop umožňuje ukládat data paralelně až do dvou SQL serverů, do cloudového a firemního (lokálního).

Cloud režim

Monitorovací systém ProCop umožňuje ukládat data do SQL databáze v cloudu přes TCP protocol HTTPS, konkrétně na server procloud.alfamik.cz port 443. Volba exportu do cloudu nemá vliv na volbu databáze pro lokální ukládání dat.

Export do cloudu má tři možnosti:

- Not Used
 - o zcela vypnuto, žádná data nejsou exportována
- Show
 - o exportovat data, ale neumožnit žádné nastavování hodnot
- Show and Control
 - o exportovat data, umožnit nastavování hodnot a ovládání dle oprávnění

🗾 Poznámka:

Provoz cloudového přístupu je licencován a je potřeba poptat u provozovatele (ALFA Mikrosystémy, spol. s r. o.)

Přepnutím volby v projektu dojde k internímu nastavení ukládání dat SQL serveru procloud.alfamik.cz a po spuštění jsou veškerá data, trendy a alarmy synchronizovány do cloudového SQL serveru.

📥 Důležité upozornění!

V případě povolení cloudového režimu jsou data dostupná uživatelům definovaným v lokálním monitorovacím projektu s příslušnými oprávněními. Pokud je povoleno i nastavení hodnot z cloudu, rovněž se i nastavování řídí oprávněními přihlášených uživatelů. Je nezbytné mít nastavena silná hesla pro přístup do systému.

浶 Tip!

Cloudový režim lze na ukázku na omezenou dobu po dohodě spustit pro malé a střední dispečinky. K zprovoznění postačuje libovolné připojení k Internetu.

SQL nastavení

V Designeru, ve Vlastnostech a dynamizacích je možné jednoduše nastavit kompletní export dat do SQL serveru stisknutím jediného tlačítka Používat lokální SQL Nastav.

Toto tlačítko nastaví následující vlastnosti, které je možné podle potřeby upravit [v hranatých závorkách předvolby]:

- Šablona řetězce připojení [Provider=SQLOLEDB.1; ...]
 - o přednastavená šablona připojení ODBC, HTTP, HTTPS, podle přístupu k SQL serveru
- Adresa serveru [localhost\procop]
 - DNS jméno nebo IP adresa serveru
 - o za lomítkem může být uvedeno jméno instance, prázdné pro default instanci
- Název databáze [ProCopDB]
 - o Název databáze, pokud se bude teprve vytvářet lze změnit
- Jméno uživatele [ProCop]
 - o Implicitní jméno uživatele, pokud se bude vytvářet databáze, lze změnit
- Heslo uživatele [m.....35]
 - o Heslo uživatele ProCop, pokud se vytvářet databáze, lze zvolit libovolně
- Zobrazovat [Pouze tento projekt]
 - Pokud více dispečinků ProCop ukládá data do stejné databáze (SQL serveru, instance), pak lze zvolit zda klienti vidí i alarmy a trendy ze všech projektů

🗾 Poznámka:

Zobrazení všech projektů - nadřazený dispečink - lze plně zobrazit pouze přes webové rozhraní ProWeb. Panel nástrojů ProCop po volbě "Zobrazovat všechny projekty" zobrazuje alarmy a trendy ze všech projektů, ale displeje pouze z aktuálního projektu. V budoucnu se plánuje i zobrazení všech displejů a dynamizací.

Oprávnění

Server oprávnění má tato nastavení:

- Počítač
 - o počítač se serverem přístupových oprávnění (DCOM), ponechat prázdné pro lokální
- Typ databáze
 - o No Export neexportovat nebude možné používat žádná oprávnění v SQL serveru
 - SQL je nutno zvolit, pokud jsou ukládány displeje, alarmy, nebo trendy do SQL, oprávnění jsou exportována do výše nastavené SQL databáze

Displeje

Nastavení displejů:

- Počítač
 - o počítač s datovým serverem (DCOM), ponechat prázdné pro lokální
- Identifikátor systému
 - o textový název systému pro rozlišení displejů ve více projektovém prostředí SQL databáze
- Typ databáze
 - No Export neexportovat nebude možné zobrazovat displeje a nastavovat hodnot z SQL serveru (nyní pouze pro ProWeb)
 - o SQL displeje a živé hodnoty jsou exportovány do SQL databáze, v současnosti slouží pro ProWeb
 - SQL Control předchozí volba rozšířená o nastavení hodnot z ProWebu, řízeno přístupovými oprávněními

Alarmy

Nastavení alarmů:

- Počítač
 - o počítač s alarmním serverem (DCOM), ponechat prázdné pro lokální
- Projekt
 - o soubor s projektem pro alarmní server
- Identifikátor systému
 - o textový název systému pro rozlišení alarmů ve více projektovém prostředí SQL databáze
- Typ databáze
 - DB alarmy postaru v BDE databázi, lze zobrazit z BDE Panelem nástrojů ProCop, ale nelze zobrazovat na webu - ProWeb
 - DB + SQL alarmy v BDE databázi, současně kopie v SQL, lze zobrazit z BDE Panelem nástrojů ProCop a zároveň z SQL pomocí ProWeb, oddělená kvitace alarmů
 - SQL alarmy pouze v SQL, Panel nástrojů ProCop i ProWeb zobrazují stejné záznamy společná kvitace
 - DB + SQL View alarmy se ukládají do BDE, ale Panel nástrojů ProCop i ProWeb zobrazují alarmy z jiných projektů v SQL - společná kvitace cizích alarmů

Trendy

Nastavení alarmů:

- Počítač
 - o počítač s trendovým serverem (DCOM), ponechat prázdné pro lokální
- Projekt
 - soubor s projektem pro trendový server
- Identifikátor systému
 - o textový název systému pro rozlišení trendů ve více projektovém prostředí SQL databáze
- Typ databáze
 - DB trendy postaru v BDE databázi, lze zobrazit z BDE Panelem nástrojů ProCop, ale nelze zobrazovat na webu - ProWeb
 - DB + SQL trendy v BDE databázi, současně kopie v SQL, lze zobrazit z BDE Panelem nástrojů ProCop a zároveň z SQL pomocí ProWeb (duplicita všech trendů)
 - o SQL trendy pouze v SQL, Panel nástrojů ProCop i ProWeb zobrazují stejné trendy z SQL
 - DB + SQL View trendy se ukládají do BDE, ale Panel nástrojů ProCop i ProWeb zobrazují trendy z jiných projektů v SQL

🗾 Poznámka:

Režim DB + *SQL View slouží zejména pro klientské pracoviště, které nahlíží do SQL databáze na alarmy a trendy, ale vlastní tam vůbec nevkládá.*

6.3 Správa serveru SQL

Tato část vlastností a dynamizací je určena ke spravovat SQL serveru, pokud databáze nebyla vytvořena při instalaci, SQL server byl nainstalován ručně a třeba později než Monitorovací systém ProCop a je potřeba vytvořit či zrušit příslušnou databázi. Všechny vlastnosti a tlačítka této části slouží pro interaktivní práci s SQL serverem, žádný parametr se neukládá. Ke správě SQL serveru potřebujeme jméno a heslo uživatele s oprávněními systémového administrátora (sa), nebo musí mít tato oprávnění aktuální uživatel Windows, pak jméno a heslo systémového administrátora nevyplňujeme.

Popis funkcí:

- Sys. Administrátor [sa]
 - o jméno systémového administrátora serveru (obvykle sa, nebo uživatel se stejnými oprávněními)
- Sys. Heslo
 - o heslo systémového administrátora SQL serveru, obvykle zadané při instalaci instance SQL serveru
- Vytvoření databáze

- tlačítko Vytvoř DB! vytvoří databázi v SQL serveru dle SQL nastavení a s vlastníkem z tohoto 0 nastavení
- uživatel (sa) musí mít oprávnění k vytváření databází, jinak operace skončí chybou 0
- v řádku označeném Výsledek operace se zobrazí, jak operace dopadla 0

Důvěryhodná instance (SET TRUS TWORTHY ON)

- stisknutím tlačítka Nastav se instance SQL serveru nastaví do režimu, kdy vlastník jedné 0 databáze může vytvářet nové, další databáze
- toto nastavení je nezbytné pro vytváření archivů trendů 53 v dalších SQL databázích 0

Opravení databáze

- stisknutím tlačítka Oprav DB! se zkontrolují některé vnitřní parametry databáze a přihlašovací 0 účty a provede se jejich oprava.
- Odstranění databáze
 - tlačítko odstraní databázi dle SQL nastavení 0
 - pro správnou funkci je nutné nejprve povolit odstranění 0
 - v tabulce označené Výsledek operace se zobrazí, jak operace dopadla 0

Zobraz stavy databází

stisknutím tlačítka Zobraz! se do tabulky označené Výsledek operace vypíšou všechny databáze nacházející se v dané SQL instanci, s jejich datem vytvoření, umístění na disku počítače a stavem.

🕭 Důležité upozornění!

Pozor, operace odstranění databáze je nevratná, SQL databáze, včetně datových souborů je ze serveru a počítače bezezbytku odstraněna!

💢 Tip!

V tabulce v řádku označeném Výsledek operace se zobrazují výsledky jednotlivých operací.

6.3.1 Obnovení, Odstranění, Změna stavu databáze

Tato část vlastností a dynamizací je logicky oddělena od Správy serveru SQL, protože jsou zde operace pro obnovení, odstranění nebo změnu stavu databáze.

Popis funkcí:

- **Povolit** operaci
 - předvolba, která musí být zatržena, aby stisk dalších tlačítek provedl danou operaci (ochrana před 0 náhodným zmáčknutím).
- Soubor pro obnovení databáze, Obnov databázi ze zálohy, Změň stav databáze
 - Funkce a práce s těmito parametry jsou popsány v následujícím odstavci Obnovení databáze. 0
- Odstranění databáze
 - tlačítko odstraní databázi dle SQL nastavení 0
 - pro správnou funkci je nutné nejprve povolit odstranění 0
 - v řádku označeném Výsledek operace se zobrazí, jak operace dopadla 0

🕭 Důležité upozornění!

Pozor, operace odstranění databáze je nevratná, SQL databáze, včetně datových souborů je ze serveru a počítače bezezbytku odstraněna!

💢 Tip!

V tabulce v řádku označeném Výsledek operace se zobrazují výsledky jednotlivých operací.

Obnovení databáze

Tento nástroj slouží pro přenos databáze z jiného SQL serveru nebo obnovení stávající databáze.

48

🗾 Poznámka

Každá SQL databáze se skládá ze dvou souborů na disku. Jeden soubor pro data s příponou .mdf a druhý soubor pro logování operací s příponou .ldf V návodu se bude jako příklad názvu databáze vyskytovat ProCopDB.

Databáze se může nacházet v několika stavech. Příklad tří stavů se kterými se setkáme při obnovení databáze.

- offline Databáze je vypnuta a její LDF a MDF soubor jsou odpojeny od SQL. Lze s nimi libovolně manipulovat. Databáze není přístupná pro SQL operace.
- online Databáze je zapnuta a její LDF a MDF soubor jsou připojeny do SQL. Nelze s nimi libovolně manipulovat. Databáze je přístupná pro SQL operace.
- restoring Databáze se obnovuje ze zálohy, musí se vyčkat změny tohoto stavu do stavu online nebo offline.

Nástroj pro obnovení databáze se nachází v SQL nastavení/Správa serveru SQL/Obnovení,odstranění, změna stavu databáze.

- 1. V menu SQL nastavení/Správa serveru SQL vyplníme údaje účtu sysadmina do SQL.
- 2. Do pole soubor pro obnovení databáze vložíme cestu k souboru se zálohou databáze s příponou .BAK. Pokud necháme cestu prázdnou, obnoví se automaticky z poslední zálohy provedené v databázi s názvem z pole SQL nastavení/Název databáze.
- 3. Databáze se obnoví s názvem z pole SQL nastavení/Název databáze. Pokud tento název databáze již existuje v SQL, připojí se k názvu **_restored** např.: ProCopDB_restored.
- 4. Povolíme operaci v poli Povolit operaci. Toto zaškrtávací tlačítko je ochrana proti náhodném spuštění některé z operací.
- 5. Stiskneme tlačítko Obnov!.
- 6. Kontrolu běžícího obnovení databáze můžeme provést v menu SQL nastavení/Správa serveru SQL/ Zobraz stavy databází tlačítkem Zobraz!. V poli Výsledek operace můžeme vidět všechny databáze v dané SQL instanci a jejich stavy (online, offline, restoring, atd.). Zde zároveň vidíme, jestli databáze již existovala a obnovuje se do např.: ProCopDB nebo ProCopDB restored.
- 7. Pokud databáze neexistuje nebo obnovujeme databázi s archivními trendy, přeskočíme tento bod. V případě že se databáze obnovila do např.: ProCopDB_restored musíme pokračovat tímto postupem.
 - a) Databáze se obnovila do stavu offline. Do stejného stavu musíme přepnout aktuální databázi v položce Změň stav databáze tlačítkem Nastav offline!
 - b) Po úspěšném přepnutí se změní text tlačítka na Nastav online!.
 - c) Stavy všech databází se dají zkontrolovat tlačítkem Zobraz! v poli Zobraz stavy databází.
 - d) Po uvedení obou databází do stavu offline, následuje prohození fyzických souborů (datový a logovací) mezi původní a obnovenou databází. V průzkumníku v umístění fyzických souborů databází, přejmenujeme aktuální soubory ProCopDB.mdf na například puvodni_ProCopDB.mdf a ProCopDB_log.ldf na puvodni_ProCopDB_log.ldf. Dále přejmenujeme obnovené databázové soubory ProCopDB_restored.mdf na původní ProCopDB.mdf a ProCopDB_log_restored.ldf na původní ProCopDB_log.ldf.
 - e) V panelu nástrojů ProCop přepneme zpátky do stavu online tlačítkem Nastav online! v menu Správa serveru SQL. Tlačítko po přepnutí do online režimu změní svůj text na Nastav offline!.
- Po obnovení databáze klikneme na tlačítko Oprav DB! pro kontrolu a případné opravení databáze po obnovení.
- 9. Funkčnost databáze si ověříme tlačítkem Zkontroluj!.

6.4 Správa databáze ProCop

Správa databáze ProCop v SQL serveru umožňuje připravit a nastavit prázdnou SQL databázi pro využití monitorovacím systémem ProCop. Automatické odstraňování alarmů 51, ředění 52 a archivace trendů 53 jsou popsány v samostatných podkapitolách.

Popis funkcí:

Očekávaná verze:

Tipy a triky

- zobrazuje očekávanou verzi SQL databáze více monitorovacích projektů může využívat jednu SQL databázi
- SQL Info Info
 - tlačítko načte základní informace o SQL serveru, výsledek a verzi serveru zobrazí v poli Výsledek operace
- Test připojení Zkontroluj!
 - o toto tlačítko se pokusí načíst veškeré informace o nastavené databázi a verzi ProCop SQL databáze
 - o v poli Výsledek operace se zobrazí aktuální a minimálně požadovaná verze
 - o vyčte se nastavení všech úloh, jako zálohování, mazání, ředění a archivace

Povolit odstranění/vyčištění

- touto předvolbou se jednorázově povolí stisknutí tlačítek Odebrat, nebo Vyčistit, jedná se o
 pojistku před náhodným stisknutím
- Odebrat tento projekt
 - pokud je předvolba Povolit odstranění/vyčištění projektu zatržena, odstraní veškerá data o tomto projektu z databáze ProCop
 - Vymazat tabulky trendů při čištění
 - o předvolba umožňuje zvolit, zda se při čištění odstraňují i historické trendy
- Vyčistit všechny tabulky
 - o volba vyčistí obsah všech tabulek, struktura databáze je zachována
 - o předvolba Vymazat tabulky trendů při čištění volí, zda se vymažou i historická data

Nastavení databáze ProCop

Následující volby jsou načteny stiskem tlačítka Zkontroluj! přímo z databáze ProCop a jsou společné pro všechny projekty v téže databázi.

🚖 Tip!

Jestliže jsou následující volby neaktivní (šedé), je potřeba vyčíst jejich stav z SQL stisknutím tlačítka Zkontroluj!

- Odstranění tabulek trendů s proměnnými [Ne]
 - při odstranění (přejmenování) proměnné z projektu je při synchronizaci odstraněna i z SQL databáze
 - tato volba definuje, se zánikem (přejmenováním) proměnné zrušena i tabulka s historickými hodnotami
 - Odstranění tabulek displejů s proměnnými [Ano]
 - při odstranění displeje se zruší i tabulka odpovídající displeji
- Alarmy: použití zákazů [Ne]
 - o globální vypínač funkce použití zákazů alarmů, společné pro všechny projekty
- Alarmy: použití kódu [Ne]
 - o globální vypínač funkce použití kódů alarmů, společné pro všechny projekty
- Alarmy: použití tříd [Ne]
 - globální vypínač funkce použití tříd alarmů, společné pro všechny projekty
- Alarmy: Automaticky kvitovat [Ne]
 - o povolení denní naplánované úlohy pro automatickou kvitaci alarmů
 - v čase [1:30:00]
 - určuje čas automatického kvitování alarmů
 - starší než [dny = 14]
 - po kolika dnech se mají alarmy automaticky kvitovat
- Alarmy: Přepočet nekvitovaných [Ne]
 - o povolení denní naplánované úlohy pro automatický přepočet počtu nekvitovaných alarmů
 - v čase [1:50:00]
 - určuje čas automatického přepočtu nekvitovaných alarmů
- Povolit denní zálohování [Ne]
 - povolení denního zálohování SQL databáze, interní naplánovaná úloha SQL databáze ProCop, vytváří dvě plné zálohy, které se střídavě přepisují

- o v čase [0:10:00]
 - určuje čas automatického denního zálohování, pokud je úloha povolena
- o počet záloh [2]
 - možnost zvolení počtu (1 až 7) záloh, které se začnou poté postupně přepisovat. Př.: Pokud zvolíme 2 zálohy, tak se uchová pondělní a úterní záloha a ve středu se přepíše pondělní záloha středeční atd...
- o do cesty [prázdné, nebo končící s \]
 - umístění záloh, standardně do podsložky Backup, nebo kam je potřeba

対 Tip!

Pokud se zálohuje do síťové cesty, je nutné aby uživatel, pod kterým běží instance SQL serveru, měl na síťové úložiště přístup. To většinou znamená, že je nutné změnit identitu instance SQL serveru ve službách.

Å Upozornění!

Jednotlivé naplánované úlohy se nemohou překrývat a provádějí se postupně. Proto je vhodné rovnou naplánovat rozdílné časy spuštění úloh a to z dostatečným odstupem.

b Upozornění!

Pozor na dostatečné místo na disku při nastavování počtu záloh.

6.4.1 Automatické odstraňování alarmů

Automatické odstraňování alarmů provádí interní naplánovaná úloha. Umožňuje odstraňovat staré záznamy dle typu a stáří podle nastavený ch parametrů.

対 Tip!

Jestliže jsou následující volby neaktivní (šedé), je potřeba vyčíst jejich stav z SQL stisknutím tlačítka Zkontroluj!

Nastavení automatického odstraňování alarmů:

- Povolit denní odstraňování alarmů [Ne]:
 - o naplánuje interní úlohu SQL databáze ProCop pro mazání alarmů dle nastavení
 - o v čase [1:10:00]
 - určuje čas spuštění naplánované úlohy, pokud je úloha povolena
- Alarmy: Automaticky odstraňovat [Ne]:
 - o povolí naplánované úloze nenávratně odstraňovat alarmy všech projektů z databáze ProCop
 - starší než [dny = 1826]
 - počet dnů, které zůstanou v databázi
- Události: Automaticky odstraňovat [Ne]:
 - o povolí naplánované úloze nenávratně odstraňovat události všech projektů z databáze ProCop
 - starší než [dny = 92]
 - počet dnů, které zůstanou v databázi

Logy: Automaticky odstraňovat [Ne]:

- povolí naplánované úloze nenávratně odstraňovat systémové logy všech projektů z databáze ProCop
- starší než [dny = 366]
 - počet dnů, které zůstanou v databázi
- Poznámky: Automaticky odstraňovat [Ne]:
 - o povolí naplánované úloze nenávratně odstraňovat poznámky všech projektů z databáze ProCop
 - starší než [dny = 3652]
 - počet dnů, které zůstanou v databázi
- Comm Logs: Odfiltrovat [Ne]
 - o přímo při příchodu odfiltruje komunikační záznamy
 - vhodné zapnout jen pro opravdu rozsáhlé projekty s řádově několika sty, až tisíci stanic
 - o komunikační záznamy se vůbec v SQL databázi nezaznamenávají => zrychlení práce s alarmy

🍐 Upozornění!

Jednou odstraněné alarmy, události, systémové události i poznámky jsou nenávratně ztraceny, pokud nebyla databáze dříve zazálohována a záloha archivována.

6.4.2 Automatické ředění trendů

Historické trendy v SQL databázi jsou ukládány v samostatných tabulkách bez omezení počtu vzorků. Limitem může být Express edice SQL databáze, nebo velikost diskového prostoru. Pokud je v tabulce vzorků hodně, například při krátké periodě vzorkování, přehledy vývoje v délce let pak mohou být velmi pomalé. Z těchto důvodů je implementován mechanizmus vícestupňového ředění a mazání vzorků trendů.

Popis mechanismu

Mechanizmus funguje tak, že naplánovaná úloha v SQL databázi nezávisle na funkci monitorovacího systému ProCop prochází jednotlivé trendy a podle nastavení vzorky starší než definovaný počet dní (obvykle jeden den, nebo od posledního spuštění) průměruje na zadanou periodu, vymazává původní vzorky a vkládá menší počet vzorků na jejich místo s delší periodou a průměrnou hodnotou. Pokud je nastaveno i odstraňování vzorků, hodnoty starší než definovaný počet dní zpětně budou nenávratně vymazány ze všech trendů.

🚖 Tip!

Jestliže jsou následující volby neaktivní (šedé), je potřeba vyčíst jejich stav z SQL stisknutím tlačítka Zkontroluj!

Nastavení automatického ředění/odstraňování trendů:

- Povolit denní rozřeďování trendů [Ne]:
 - o naplánuje interní úlohu SQL databáze ProCop pro ředění/odstraňování trendů dle nastavení
 - v čase [2:10:00]
 - určuje čas spuštění naplánované úlohy, pokud je úloha povolena
- 1. Povolit rozředění [Ne]:
 - na periodu [min = 10 minut]
 - požadovaná minimální perioda trendů po prvním rozředění, trendy s touto a delší periodou se neředí
 - starší než [dny = 31]
 - počet dnů od současnosti, které nebudou prvním ředěním dotčeny = zůstanou v původních periodách
 - 2. Povolit rozředění [Ne]:
 - na periodu [min = 30 minut]
 - požadovaná minimální perioda trendů po druhém rozředění, trendy s touto a delší periodou se neředí
 - starší než [dny = 92]
 - počet dnů od současnosti, které nebudou druhým ředěním dotčeny = zůstanou v původních periodách
 - 3. Povolit rozředění [Ne]:
 - na periodu [min = 1 h]
 - požadovaná minimální perioda trendů po rozředění, trendy s touto a delší periodou se neředí
 - starší než [dny = 366]
 - počet dnů od současnosti, které nebudou třetím ředěním dotčeny = zůstanou v původních periodách
- 4. Povolit rozředění [Ne]:
 - na periodu [min = 6 h]
 - požadovaná minimální perioda trendů po čtvrtém rozředění, trendy s touto a delší periodou se neředí
 - starší než [dny = 731]
 - počet dnů od současnosti, které nebudou čtvrtým ředěním dotčeny = zůstanou v původních periodách
- Povolit odstranění [Ne]:
 - pro všechny vzorky starší než [dny = 1826]

- počet dnů od současnosti, které zůstanou v databázi nedotčeny, starší vzorky budou nenávratně odstraněny
- Rychlé ředění [Ne]:
 - pro velké databáze s řádově desítkami tisíc trendů je možné zapnout rychlé ředění, které nezkoumá skutečné periody trendů
 - používá se k ředění nastavená perioda, algoritmus ředění je pak výrazně rychlejší, ale nedokonale ředí trendy se změnou periody

🕭 Upozornění!

Jednou rozředěné či odstraněné vzorky trendů jsou nenávratně ztraceny, pokud nebyla databáze dříve archivována, nebo zálohována. Z rozředěných vzorků tak vzniká aritmetický průměr odstraněných, který má dobrou vypovídací hodnotu o stavu technologie, potírá však krátkodobé extrémy (špičky).

6.4.3 Archivace trendů

Mechanismus archivace trendů má několik důvodů, proč byl vyvinut a může být užitečné jej využít. Historické trendy v SQL databázi jsou ukládány v samostatných tabulkách bez omezení počtu vzorků. Limitem může být Express edice SQL serveru, nebo velikost diskového prostoru. Pokud je v tabulce vzorků hodně, například při krátké periodě vzorkování, přehledy vývoje v délce let pak mohou být velmi pomalé. Využití mechanizmu ředění trendů 52 může být nežádoucí, nebo nedostatečně omezovat velikost. Pro tyto případy je vhodné použít archivaci trendů do dalších SQL databází.

Jak archivace funguje

Naplánovaná úloha v čase spuštění vyhodnocuje podle nastavených parametrů, které vzorky za kolik dni přesune z průběžné databáze do archivní.

- K vytváření archivních databází musí být instance SQL databáze důvěryhodná 47 (SET TRUSTWORTHY ON)
- Archivační úloha po spuštění vyhodnotí, zda jsou v průběžné databázi vzorky starší než zadaný limit (x měsíců)
- Ověří existenci archivní databáze, nebo vytvoří novou, pojmenuje stejně jako původní s připojením _HTyyyymmdd
- Vytváří jednotlivé tabulky trendů v databázi, **plní je hodnotami z průběžné databáze, kde je** současně odmazává
- Kontroluje maximální velikost databáze, pokud je omezena a přepíná archivy (12 -> 6 -> 3 -> 1 měsíců), pokud se data nevejdou
- Může vytvořit databázi i časem mimo nastavený rozsah, pokud velikost překročí omezení a následně přepne rozsah na kratší
- Prohlížeč trendů v Panelu nástrojů ProCop **zobrazuje hodnoty ze všech databází současně**, pokud je to potřeba
- Zálohování databáze ProCop zálohuje průběžnou databázi i archivní databáze, pokud v nich byla provedena změna (nemají příznak jen pro čtení)
- Archivní databáze jsou po po zálohování **nastaveny jen pro čtení**, pouze archivační úloha může povolit i zápis, pokud je potřeba
- Archivní databáze lze ze serveru kdykoliv odstranit, případně znovu doplnit, či obnovit ze záloh, v monitorování budou opět k dispozici

💢 Tip!

Jestliže jsou následující volby neaktivní (šedé), je potřeba vyčíst jejich stav z SQL stisknutím tlačítka Zkontroluj!

Nastavení automatické archivace trendů:

- Povolit denní archivaci trendů [Ne]:
 - o naplánuje interní úlohu SQL databáze ProCop pro archivaci trendů dle nastavení
 - o v čase [3:10:00]
 - určuje čas spuštění naplánované úlohy, pokud je úloha povolena
- Rozsah archivu [12 měsíců]:
 - o určuje délku archivů (databází), do kterých budou přesouvány starší vzorky
 - o rozsah lze nastavit na 1 měsíc, 3 měsíce, 6 měsíců a 12 měsíců

- Archivovat vzorky starší než (0=auto) [měsíc]:
 - o v průběžné databázi bude udržován nastavený počet měsíců vzorků trendů zpětně od současnost
 - volba 0=auto znamená automaticky stejný počet měsíců, jako je rozsah archivu, tedy 1 měsíc, 3 měsíce, 6 měsíců a 12 měsíců, dle nastavení druhého parametru
 - o význam volby 0=auto je zřejmý z omezení velikosti databáze
- Max. velikost (0=vyp) [MB=8000]
 - volba udává maximální velikost databází (archivů i průběžné databáze), je vhodné ji z bezpečnostních důvodů kontrolovat při použití edice SQL Express (max. 10GB)
 - pokud je velikost archivu překročena, je dojde k automatickému zkrácení Rozsahu archivu na kratší, pokud je je volba Archivovat vzorky starší než = 0 auto, dojde k přepnutí i velikosti průběžné databáze
 - o pokud je překročena velikost průběžné databáze (po archivaci), zkrátí se ještě o měsíc
 - \circ 0 = vyp. znamená, že se velikost vůbec nekontroluje obvykle pro SQL servery Standard
- Max. čas běhu (0=vyp):
 - kontroluje se maximální doba běhu archivace, pokud překročí nastavenou dobu, tak se přeruší a pokračuje se další den
 - o slouží hlavně k omezení zatížení SQL serveru po změně (zapnutí) nastavení archivace
- Archivovat po [den = 7]
 - o limit archivace vzorků na jeden průchod
 - pokud se archivuje pravidelně každý den, pak se toto omezení neprojeví (archivuje se právě jeden den)
 - pokud dojde ke zapnutí / změně archivace, pak se současně zpracovává nastavený počet dnů vzorků v postupně ve všech trendech
 - po dokončení posledního trendu, se zkontroluje maximální doba běhu a archivuje se další nastavený počet dní v každém trendu

6.4.4 Reindexace a Shrink

Reindexace

V tabulkách v SQL databázi dochází vlivem ředění trendů, archivování databází a dalších operací k mezerám ve fyzických datech na disku. Do těchto mezer jsou pak vkládána nová data a jejich logické pořadí se neshoduje s jejich fyzickým uspořádáním na disku. Z tohoto důvodu mohou být operace nad těmito tabulkami pomalejší.

Reindexace je operace nad SQL tabulkami, která přeskládá všechny záznamy v tabulce tak, aby fyzické uspořádání na disku odpovídalo jejich logickému pořadí v SQL tabulce. Tím se stanou jednotlivé operace jako např.: hledání trendů, přepínání displejů rychlejšími. Touto operací nedojde ke zmenšení databáze, ale naopak může dojít k jejímu zvětšení. Reindexace nemaže, ani nijak neupravuje uložená data, pouze jejich vnitřní uspořádání.

Shrink

Pokud je potřeba zmenšit fyzickou velikost databáze na disku, je možné použít operaci shrink. Pomocí této operace se přeskládají vnitřní data databáze tak, aby zabírala co nejmenší prostor na disku. Shrink data seřadí stylem přesunu posledního záznamu na první volné místo. Po tomhle seřazení dochází ke zhoršení výkonu operací nad databází, proto je vhodné provádět shrink jen ve výjimečných případech, například po smazání velkého množství dat, odsunu dat do archivních databází nebo nedostatečného místa na disku.

🚖 Tip!

Jestliže jsou následující volby neaktivní (šedé), je potřeba vyčíst jejich stav z SQL stisknutím tlačítka Zkontroluj!

Reindexace a shrink

- Povolit reindexaci a operaci shrink - povolení automatické naplánované úlohy
 - Ano: naplánuje interní úlohu SQL databáze ProCop pro reindexaci a operaci shrink dle nastavení
 - o Ne: úloha nebude naplánována
- Typ úlohy
 - Shrink: úloha provede pouze operaci shrink
 - Reindexace: úloha provede pouze reindexaci
 - Shrink a Reindexace: bude provedeno obojí
- Den v týdnu
 - o volba naplánované úlohy jednou týdně (volba dne v týdnu), každý den nebo jednou měsíčně
 - Den v měsíci
 v případě volby jednou měsíčně je možnost zvolit den v měsíci
- V čase
 - o čas provedení naplánované úlohy

V dalším podmenu se nachází podrobnější nastavení minimálních hodnot pro provedení reindexace nad tabulkou.

Nastavení minim reindexace

- Počet stránek tabulky
 - minimum počtu stránek tabulky, aby byla nad ní provedena operace reindexování. Doporučené minimum je 1000 stránek tabulky
- % fragmentace pro přestavení
 - přestavění indexu spočívá ve smazání tabulky a vytvoření úplně nové. Fragmentace v procentech udává míru mezer ve fyzických datech na disku. Doporučené minimum je 30%.
- % fragmentace pro reorganizaci
 - reorganizace indexu pouze upravuje stávající uspořádání dat bez mazání a znovuvytváření tabulky. Doporučené minimum je 5%.

🞽 Poznámka

Reindexace ani operace shrink nijak nemění ani nemaže data uložená v databázi, pouze mění jejich uspořádání na disku počítače. Výchozí nastavení je provádění operace shrink a reindexace jednou měsíčně v prvním dni měsíce v 22 hodin a 10 minut.

6.4.5 Nástroje

Nástroje

V tomto podmenu je k dispozici několik databázových úloh pro práci s databází. Názvy úloh i jejich výsledky jsou uvedeny pouze v anglickém jazyce. Úlohy jsou rozděleny na synchronní a asynchronní a dále se dělí do několika kategorií podle jejich zaměření.

Synchronní úlohy

Tyto úlohy okamžitě vrátí výsledek operace do tabulky v řádku Výsledek operace.

KILL:

- Running task
 - o ukončí probíhající naplánovanou úlohu

Reindexace a shrink Povolit reindexaci a operaci shrink Ano • • Typ úlohy Shrink a Reindexace Den v týdnu Jednou měsíčně • Den v měsíci 1 V čase 22:10:00 Nastavení minim reindexace Počet stránek tabulky 1000 % fragmentace pro přestavení 30 % fragmentace pro reorganizaci 5 Reindexace a shrink

RUN:

• Null UCID

pokud nastane alarm: Nemohu se připojit: unikátní ID počítače se změnil nebo se připojujete z jiného počítače! Je možno toto unikátní číslo počítače (UCID) vynulovat, systém si pak sám vytvoří UCID nový

SHOW:

- Backup or Restore progress
 - o pokud probíhá zálohování nebo obnovení databáze zobrazí jeho stav a odhadovaný čas dokončení
- Database size
 - zobrazí velikost ProCop databáze
 - Databases info
 - o zobrazí v SQL instanci název, datum vytvoření a stav všech databází
- Scheduler log
 - zobrazí log naplánovaných úloh
- Scheduler tasks
 - o zobrazí časy naplánovaných úloh a název probíhající úlohy

Asynchronní úlohy

0

Tyto úlohy mají delší dobu trvání, proto jsou spuštěny v plánovači úloh. Jejich spuštění a průběh lze sledovat pomoci nástroje **SHOW: Scheduler log**. Jejich časy naplánování lze zjistit pomocí **SHOW: Scheduler tasks**.

DELETE: slouží pro uvolnění místa v SQL databázi

- Logs older than year
 - smaže z databáze logy starší než jeden rok
- Not used trend tables
 - o smaže nepoužívané tabulky historických trendů, které nejsou navázané na proměnné

RUN: spouští okamžitě naplánované úlohy popsané v předchozích kapitolách. Pokud je úloha zároveň nastavena jako naplánovaná, tak se její naplánování nijak nezruší ani nezmění.

- Alarm auto accept
 - spustí úlohu denního automatického kvitování alarmů popsanou v kapitole Správa databáze ProCop
 49
- Alarm auto recalculation unaccept
 - spustí úlohu automatického přepočtu nekvitovaných alarmů popsanou v kapitole Správa databáze
 ProCop 49
- Backup
 - o spustí úlohu zálohování SQL databáze popsanou v kapitole Správa databáze ProCop 49
 - Restore DB to *_restored
 - obnoví zálohu databáze ProCop do ProCop_restored popsanou v kapitole Obnovení, Odstranění, Změna stavu databáze 481
- Shrink and Reindex
 - o spustí úlohu reindexace a operace shrink popsanou v kapitole Reindexace a shrink 54
- Trends rarefy
 - o spustí úlohu ředění trendů popsanou v kapitole Automatické ředění trendů 52

7 Diagnostika zařízení

V této části jsou uvedeny návody, jak zjistit příčinu poruchy komunikace zařízení, jako jsou AlfaBox 57, či AlfaMeter 58.

7.1 AlfaBox

AlfaBox připojen externím modemem přes GPRS

Základní kroky diagnostiky:

- V monitorovacím systému ProCop otevřeme Řízení modulů, nebo také kombinací kláves ALT+M a spustíme komunikaci tlačítkem Start
 - AlfaBox komunikuje, zvyšují se přenesená data a v některém okamžiku, např. při přenosu alarmu, nebo trendů vyhlásí poruchu
 - provedeme vzdálený reboot koncentrátoru (cca 2 3 minuty) a zkusíme znovu
 - pokud nelze reboot provést, nebo se situace nezlepší, kontaktujeme servis
 - AlfaBox píše "Přihlašuji se...", načítá se čas vypisuje Opakuji <X> a nakonec se objeví stav "Porucha!"
 - pro komunikaci po síti (Ethernet UDP, APN) zjistíme IP adresu a ze serveru ověříme konektivitu nástrojem PING IPAdresa
 - pokud není na PING odezva, a jiné AlfaBoxy se stejným způsobem připojení komunikují, budeme nutná diagnostika na místě
 - nekomunikují li všechny zařízení se stejným způsobem připojení, prověříme síťově připojení, kontaktujeme servis
 - při jiné, např. sériové komunikaci s koncentrátorem prověříme komunikační cestu, ovladače sériového portu PC, kartu, převodník a kabeláž
- 2. AlfaBox s GPRS modemem namontován v technologii
 - o prověříme napájení AlfaBoxu, zelená LED PWR pod displejem musí trvale svítit
 - pokud nesvítí, prověříme, že je zdroj v zásuvce, zásuvka pod napětím, můžeme změřit voltmetrem napětí na konektoru jack 5V DC
 - pokud zelená PWR LED nesvítí vyměníme zdroj AlfaBoxu za nový
 - pokud zelená LED PWR svítí, zkontrolujeme červenou LED a displej
 - cca do 1 minuty po připojení napájení / resetu AlfaBoxu svítí trvale červená LED a píše AlfaBox
 3.x OS Booting...
 - pokud červená LED svítí trvale i po 1 minutě, nápis OS Booting nezmizel, nebo se zařízení samo rebootuje
 - vy měňte napájecí zdroj za nový a opakujte postup diagnostiky, pokud toto nepomohlo demontujte zařízení a s původním zdrojem zašlete na opravu
 - umíte-li nakonfigurovat náhradní AlfaBox, přesuňte komunikační modulu (nezměnit pozice!), přepojte kabely do stejný ch portů a znovu vyzkoušejte
 - na displeji se zobrazuje nápis ProBox 3.X.Y.Z, zelená PWR svítí, žlutá led WDT pravidelně bliká
 - červená LED může pravidelně blikat, nesmí trvale svítit (blikání indikuje nějaké alarm, který neodeslal ještě na dispečink)
 - tlačítky (modré šipky) nahoru a dolů lze procházet alarmy a události
 - Alarm: Chyba při práci s modemem indikuje chybu modemu, nebo chybně nastavenou komunikační rychlost modemu
 - prověříme napájení externího modemu, obvykle modrá LED externího modemu musí

blikat

- o externí modem nijak nesvítí, nebliká a je připojen do funkční zásuvky
 - prověříme funkčnost zásuvky, prověříme připojení sériové linky modemu Cannon DB9 do konektoru RJ45 AlfaBoxu
 - změříme napájení na konektoru modemu dle štítku zdroje
 - vyměníme zdroj modemu
 - vyměníme modem (přesuneme SIM)
- **interní modem** by měl být umístěn v COM4 AlfaBoxu, přes otvor v čelním krytu by měl být vyveden kabel antény
 - má dvě LED u levého RJ45 konektoru COM4, zelená LED indikuje napájení modemu, oranžová LED indikuje stav a chování stejně, jako LED externího modemu
 - pokud zelená LED nesvítí (cca po 1 minutě od resetu zařízení), vyměníme modemový modul AlfaBoxu, přesuneme SIM
- následující chování indikační LED modemu (modré, oranžové) modemu je obdobné u externího i interního modemu
 - pravidelně bliká 50% (cca 0,5s nesvítí, pak cca 0,5s svítí) indikuje nemožnost modem přihlásit do GSM sítě operátora
 - přechodový jev, cca max 1 m po startu by se měl stav změnit
 - vadná, špatně vložená, neaktivní, zablokovaná SIM karta ověříme funkčnost SIM karty v jiném zařízení
 - odpojená / utržená anténa, zcela nedostatečný GSM signál prověříme jiným zařízením se SIM stejného operátora sílu signálu, vyměníme anténu
 - pravidelně bliká 10% (tzv jednou krátce, cca 0,9s nesvítí, pak cca 0,1s blikne) indikuje přihlášení do sítě GSM, není však připojení do datové sítě GPRS
 - přechodový jev, cca max 1 m po přihlášení do GSM sítě by se měl modem přihlásit do sítě GPRS a změnit blikání
 - trvale bliká jednou krátce, na SIM není aktivován přístup do požadovaného APN, špatná, nebo nedokončená konfigurace tarifu SIM u operátora
 - pravidelně bliká dvakrát krátce cca 0,9s nesvítí, pak dvakrát krátce blikne), nebo nahodile pravné přihlášení do GPRS sítě
 - přenos dat rozbliká při komunikaci LED zcela nahodile, v klidovém stavu, když se data nepřenášejí bliká 2x krátce s delší mezerou
 - toto je požadovaný stav, na displeji AlfaBoxu by neměl být alarm, nebo i jeho odhlášení "OK: Chyba při práci s modemem"
 - z dispečinku by mělo být možné z řízení modulů navázat komunikaci
 - pokud nelze navázat komunikaci, ověřte konektivitu pomocí PING, kontaktujte servis

7.2 AlfaMeter

AlfaMeter - připojen přes GSM/SMS

Základní kroky diagnostiky:

- 1. V monitorovacím systému ProCop otevřeme Řízení modulů, nebo také kombinací kláves ALT+M
 - o Najdeme příslušnou GSM cestu a ověříme, že modem je v pořádku nevypisuje žádnou chybu
 - pokud modem vypisuje chyby, např. Modem Error, Baud Rate, prověříme připojení k modemu a modem
 - následující chování indikační LED modemu (modré, oranžové) modemu je obdobné u externího modemu i interního modemu v AlfaPortu
 - pravidelně bliká 50% (cca 0,5s nesvítí, pak cca 0,5s svítí) indikuje nemožnost modem přihlásit do GSM sítě operátora
 - přechodový jev, cca max 1 m po startu by se měl stav změnit
 - vadná, špatně vložená, neaktivní, zablokovaná SIM karta ověříme funkčnost SIM karty v jiném zařízení
 - odpojená / utržená anténa, zcela nedostatečný GSM signál prověříme jiným zařízením se SIM stejného operátora sílu signálu, vyměníme anténu

- pravidelně bliká 10% (tzv jednou krátce, cca 0,9s nesvítí, pak cca 0,1s blikne) indikuje přihlášení do sítě GSM
 - tento stav by měl být až na konfigurační spojení s AlfaM eterem trvalý, modem je připraven ke spojení, přijímá SM S
- trvale svítí 100% indikuje datový hovor (obdoba telefonního hovoru), probíhá přenos dat s jiným modemem v síti GSM 2G
 - spojení s AlfaM eterem je pouze pro konfiguraci a probíhá v čase, kdy je protistrana napájená (nespí) a připojena do GSM sítě
 - u bateriových verzí probíhá dle konfigurace, trvale napájené verze kdykoliv pozměně konfigurace
- modem je v pořádku module Meter (AlfaMeter) by měl vypisovat "Pauza" a datum / čas poslední komunikace
 - pokud by la provedena změna konfigurace a zařízení je dostupné, může probíhat komunikace = datový hovor
- AlfaBox píše "Přihlašuji se...", načítá se čas vypisuje Opakuji <X> a nakonec se objeví stav "Porucha!"
 - pro komunikaci po síti (Ethernet UDP, APN) zjistíme IP adresu a ze serveru ověříme konektivitu nástrojem PING IPAdresa
 - pokud není na PING odezva, a jiné AlfaBoxy se stejným způsobem připojení komunikují, budeme nutná diagnostika na místě
 - nekomunikují li všechny zařízení se stejným způsobem připojení, prověříme síťově připojení, kontaktujeme servis
 - při jiné, např. sériové komunikaci s koncentrátorem prověříme komunikační cestu, ovladače sériového portu PC, kartu, převodník a kabeláž
- 2. V monitorovacím systému ProCop otevřeme Displeje, nebo také kombinací kláves ALT+D a dále displej s přehledem AlfaM eterů měřičů
 - o v řádku měřiče jsou postupně následující informace
 - stav komunikace, vyřazení z komunikace
 - AlfaMeter je právě vzhůru možnost navázání spojení a vyčtení dat
 - Chyba komunikace s měřičem M-Bus Měřič neodpovídá AlfaMeteru na dotazy chyba v kabeláži, měřič nemá nastavenu žádnou primární adresu (stačí jakákoliv nenulová)
 - Poslední SMS dorazila indikace, že v poslední naplánované době SMS dorazila
 - AlfaM eter neposílá SMS poslední SMS nedorazila i když byla očekávána je potřeba zkusit navázat spojení a pokud to není možné, prověřit AlfaM eter na místě
 - stav GSM signálu v procentech, pokud je menší než 20% je nutné lépe umístit anténu pro spolehlivý přenos
 - potřeba volání na AlfaMeter indikuje, že bude v příští možné době navázáno datové spojení s AlfaMeterem
 - potřeba inicializace AlfaM eteru indikuje, že by la provedena změna konfigurace, ale ještě neby la nahrána do AlfaM eteru (nebo navázáno spojení)
 - síťový AlfaMeter na batérii výpadek napájení AlfaMeteru, je napájen batérií nefunkční, nebo odpojený zdroj, výpadek fáze
 - baterie je vybitá je nutné vyměnit baterii
 - napájení z batérie / ze sítě zobrazuje typ napájení při první inicializaci
 - napětí baterie přesné napětí baterie v AlfaMeteru
 - parametry, nastavení, poslání SMS
 - pokud AlfaMeter vykazuje nějakou chybu, jako nekomunikuje, měřič nekomunikuje, slabá baterie, bude nutné AlfaMeter prověřit na místě

3. Kontrola AlfaMeteru na místě

- má-li AlfaMeter připojen napájecí zdroj, svítí nepřetržitě zelená LED
 - pokud nesvítí, prověříme, že je zdroj v zásuvce, zásuvka pod napětím, můžeme změřit voltmetrem napětí na konektoru jack 5V DC
 - pokud zelená LED stále a zdroj je připojený, vyměníme zdroj AlfaM eteru za nový
- bateriový AlfaMeter z důvodu úspory energie LED vypíná, svítí pouze v době, kdy je vzhůru (odečítá měřič, odesílá SM S, čeká na konfiguraci)
 - stiskem delšího tlačítka Function probudíme AlfaMeter, zelená LED by měla blikat

- významy LED a funkce tlačítek jsou vyobrazeny na nálepce a popsány v montážním letáku, počet bliknutí indikuje stav dle dokumentace
- nelze-li AlfaM eter tlačítkem Function zapnout, napětí baterie je vyšší, než 4 V a baterie je správně připojená, kontaktujte servis
- o pokud LED blikají, spočítejte počet bliknutí červené LED
 - 0 = vše v pořádku
 - 1 = chyba modemu, kontaktujte servis
 - 2 = SIM, ověřte funkčnost SIM v jiném zařízení, u operátora
 - 3 = PIN, zrušte na SIM kartě požadavek na PIN v jiném zařízení
 - 4 = nelze se registrovat v síti, ověřte signál (počet bliknutí modré LED alespoň 2), přesuňte anténu, ověřte funkčnost SIM v jiném zařízení, u operátora
 - 5 = SMS nelze odesílat / přijímat SMS, ověřte posílání SMS s touto SIM v jiném zařízení, u operátora
- o prověřte spojení s dispečinkem, nechte navázat spojení a vyčíst data
 - pokud toto nelze, nebo zařízení vyměňujete:
 - vložte do nového AlfaMeteru původní SIM, která identifikuje konkrétní umístění / měřič / AlfaMeter
 - současně stiskněte tlačítko Function a Reset
 - AlfaMeter provede test GSM modemu, SIM karty, síly signálu, připojeného M-Bus zařízení, napájecího zdroje a připraví se na přijetí konfigurační SMS
 - o zašlete z dispečinku konfigurační SMS
 - pokud je konfigurace úspěšná, zařízení zašle zpět vyčtená data z M-Bus zařízení a stavové informace (napětí baterie, sílu signálu)

• Výměna baterie AlfaMeteru

- pokud je signalizováno na dispečinku, že je baterie vybitá, je nutné provést její výměnu
- výměnu baterie je možné provádět i se zapojeným síťovým zdrojem
- životnost baterie je závislá na četnosti odečtu, datového spojení a síle signálu, typicky delší než jeden rok
- nový stav napětí baterie se zobrazí na dispečinku až po navázání komunikace a vyčtení dat, nebo zaslání SM S

8 Bezpečné nastavení hodnoty

Bezpečné hromadné nastavení hodnot do technologie se může jevit jako jednoduchý problém. Jednoduchá řešení mohou být funkční, nic méně může za jistých okolností dojít hromadnému nastavení i v době, kdy toto nepožadoval uživatel. Následující odstavce budou věnovány složitějšímu, ale bezpečnému řešení pro případy, kdy tato nastavení smí být prováděny **pouze na pokyn uživatele**.

Tento postup zabezpečuje, aby nedošlo nechtěnému odeslání hodnoty do cílové proměnné bez vědomí uživatele v případech kdy:

- nastavení nebylo na povel uživatele
- zdrojová proměnná neby la nastavena
- zdrojová a cílová proměnná se shodují (zabrání dvojímu nastavení stejnou hodnotou)

K tomuto účelu je potřeba využít modulu paměťové databáze Memory Database 77, dynamizace OnCommand uživatelsky odeslané z technologického displeje

Å Důležité upozornění!

Pro hromadná nastavení může postačovat řešení, kdy uživatel nastavuje přímo hodnotu pomocné proměnné například z paměťové databáze a ta je synchronizována dynamizacemi např. při změně lf Changed, nebo cyklicky Cyclic dynamizací násobným přiřazením hodnoty SetValue, nebo skriptem do technologických proměnných.

Pokud však dojde ke změně proměnné bez zásahu uživatele (po restartu, upgradu, nekorektním vypnutí systému, sloučením projektu, obnovením ze zálohy, atp.) může dojít k návratu do staršího, nebo iniciálního stavu proměnné a jejímu automatickému nastavení do všech cílových technologických proměnných ihned po spuštění monitorovacího projektu a to bez vědomí uživatele. **Pokud je potřeba tomuto zabránit, je nutné použít v těchto kapitolách popisovaný postup**.

Koncepce bezpečného hromadného nastavení hodnoty

Aby nedošlo k nepotvrzené změně cílových hodnot, nelze použít dynamizace globální dynamizace datového serveru jako Cyclic, If Changed, If True, If False, If Changed, kterou systém vyhodnocuje automaticky 1x za sekundu mohou být závislé na stavu nějaké proměnné.

K nastavení hodnoty použijeme dynamizaci OnCommand. Tato dynamizace funguje tak, že uživatel z technologického displeje po stisku příslušné volby odešla z klienta na datový server zprávu, označenou příslušným textovým identifikátorem.

Dynamizace OnCommand v datovém serveru (v dynamizacích některého modulu) se tedy provede jedině tehdy, pokud uživatel příkaz se stejným textovým identifikátorem odeslal.

Aby chom předešli náhodnému odeslání příkazu uživatelem, zdvojíme bezpečnost tak, že hodnotu, kterou budeme hromadně nastavovat do cílových technologických proměnných nastavíme po startu systému a po každém hromadném nastavení na zakázanou hodnotu např. *Nezadáno>*. K tomuto účelu použijeme modul paměťové databáze⁷⁷, kde si zadefinujeme diskrétní paměťovou proměnnou. Pomocí skriptu pak ověříme, že zdrojová proměnná není v nezadaném stavu, provedeme příslušná nastavení a ihned ji do nezadaného stavu z bezpečnostních důvodů opět vrátíme. Umožníme také uživateli kdykoliv nastavit nezadaný stav, pokud si hromadné nastavení rozmyslí.

💢 Tip!

Příklad bezpečného hromadného nastavení hodnot naleznete v následující podkapitole a ukázkovém projektu na v sekci ke stažení, ProCop 3.7 pod názvem Tips and Tricks Sample 1.

8.1 Příklad hromadného nastavení

Na jednoduchém příkladu budou uvedeny postupy pro nastavení hodnoty zdrojové proměnné, její kontroly a následného odeslání do cílové proměnné.

🚖 Tip!

Příklad bezpečného hromadného nastavení hodnot naleznete v sekci ke stažení, ProCop 3.7 pod názvem Tips and Tricks Sample 1.

Vložení modulu, proměnné a její nastavení

V Panelu nástrojů ProCop otevřeme v režimu Designer Správu projektu, vybereme Cesty a moduly vložíme modul Memory Database.

V položce Proměnné právě vytvořeného modulu vytvoříme zdrojovou proměnnou.

Ve vlastnostech a dynamizacích proměnné nastavíme parametry podle obrázku.

🗾 Poznámka:

Význam jednotlivých parametrů je uveden v kapitole Memory Database

🍕 Vlastnosti a dynamiz	ace	₽
	Hodnota	
Název	MD.SourceValue	
🖩 Тур	Discrete	•
🗉 Popis	Zdrojová hodnota	
🗉 Pole	1	
🗉 Použito	Ano>	
💷 Inicializace	0	
💷 Ukládat poslední hodnotu	Ne	•
Ukládat do inicializace (pio)	Ne	Ŧ
🗆 Limity, přepočty		
💷 Dolní mez	0	
💷 Horní mez	2	
Násobitel	1	
💷 Posun	0	
💷 Jednotka		
💷 Formát výpisu	<user table=""></user>	
🗉 Tabulka textů	0: <nezadáno> 1:Vypnuto 2:Zapnuto</nezadáno>	•••
Styl záznamu	Set Value	
🗄 Alarm		

Vlastnosti a dynamizace zdrojové proměnné

V sekci Limity a přepočty v položce Formát výpisu zvolíme <User Table> V položce Tabulka textů klikneme na tři tečky a vložíme tabulku stavů, například:

- <Nezadáno> (0)
- Vypnuto (1)
- Zapnuto (2)

V příkladu Tips and Tricks Sample 1 jsou vytvořeny tři cílové binární proměnné se stavy:

- Vypnuto (0)
- Zapnuto (1)

🗾 Poznámka:

Cílových proměnných může být libovolný počet. Také se nemusí jednat jen o binární proměnné. Stavů v pomocné

zdrojové proměnné bude vždy o jeden navíc, iniciální.

Vytvoření dynamizace pro zadání zdrojové a kopírování do cílových proměnných

Pro nastavení, resetování a odeslání proměnné si vytvoříme dynamizaci lokální nabídka na libovolné entitě, například na tlačítku poi stisknutí levého tlačítka myší. V lokální nabídce umožníme nastavit hodnotu paměťové proměnné, resetovat tuto hodnotu do stavu *Nezadáno>* a pokud je tato proměnná zadána, pak odeslat příkaz se shodným textem na datový server a spustit tak dynamizaci OnCommand.

Další obrázek je displej z ukázkového projektu Tips and Tricks Sample 1.



Tips and Tricks Sample 1, Display 1 - Odeslání hodnoty

Ve vlastnostech a dynamizacích tlačítka vložíme v dynamizace na akci myší novou položku s těmito parametry:Typ podmínky:On MousePodmínka:Left DownTyp akce:Floating Menu

V lokální nabídce přidáme např. tři řádky nové řádky (případně s oddělovačem pro přehlednost), jak je vidět na obrázku. Tyto umožní nastavení a resetování zdrojové hodnoty a odeslání příkazu pro spuštění skriptu hromadného nastavení. Pro přehlednost jednotlivé řádky zneviditelňujeme dle toho, jak jsou potřeba. Pokud není zadána zdrojová hodnota, žádné další řádky nemají smysl a jsou neviditelné. Jakmile je hodnota zadána, zobrazuje se možnost jejího resetování, nebo zaslání do technologie.

🔃 Vlastnosti a dynamiz	ace		Ţε
Typ podmínky	Podmínka	Typ akce	
🗎 Vlastnosti entity			
Dynamizace vlastností			
🔅 Dynamizace animační			
🔾 Dynamizace na akci myší			
🚫 On Mouse 🛛 🗵	Left Down 🗵	Floating Menu	
🗂 Menu	-	-	
🔊 Resetuj zdrojovo	Řádek 🔳	Set Value	
Nastav zdrojovo	Řádek 🗵	Set Value Dialog	
چا -	Oddělovač 🗵	-	
Odešli zdrojovou	Řádek 🗵	Send Command	-
	Hodnota		
Název položky	Nastay zdrojov	/ou hodnotu	
Tvp položky	Řádek		-
Aktivní (výraz)	true		
Viditelný (výraz)	MD.SourceValu	ie = 0	***
Test správnosti zadání		Kompilovat!	
Výsledek kompilace			
Typ akce	Set Value Dialo	g	-
Parametry akce			
🗉 Text dialogu	<zdrojová hod<="" td=""><td>Inota></td><td></td></zdrojová>	Inota>	
Proměnná. Vlastnost[x]	MD.SourceValu	ie	•••
🗉 Тур	Discrete		•
🗉 Min	0		
🖩 Max	2		
Načíst předvolby		Načíst	
Předvolby	1:Vypnuto 2:Z	apnuto	•••
🗆 Řádek00			
Popis	Vypnuto		
Přiřadit hodnotu	1		
🗆 Řádek01			
Popis	Zapnuto		
Přiřadit hodnotu	2		
	1 Y'1 C . TT	1 D 1	

Dynamizace tlačítka Set Value Dialog

Tlačítkem Načíst lze z formátu výpisu proměnné v databázi načíst a upravit předvolby v nastavovacím dialogovém okně. Můžeme odstranit volbu <Nezadáno>, jelikož ji umožňujeme resetovat zvláštní dynamizací.

Poslední řádek odešle dynamizací Send Command textový identifikátor na server, kde se provede dynamizace OnCommand se stejným identifikátorem. Tento postup zajistí, že se nikdy neodešle příkaz a nespustí nastavovací skript bez vědomí uživatele, pouze po stisknutí tohoto tlačítka a výběru "Odešli zdrojovou hodnotu".

V položce Aktivní výraz můžeme zadat podmínku zdrojové hodnoty větší než nula, která pokud nebude splněna, zneaktivní volbu odeslat hodnotu. Tímto zamezíme odeslání nezadané zdrojové hodnoty do cílové. Na obrázku vidíme, jak vypadá nesplněná podmínka v akci.

💢 Tip!

Pokud podmínky u jednotlivých řádku namísto do vlastnosti Viditelný (výraz) zadáme do vlastnosti Aktivní (výraz), a do Viditelný (výraz) zadáme hodnotu true, všechny řádky budou vždy viditelné a ty neplatné budou šedé a bez možnosti je stisknout. Tento přístup může být pro některé uživatele přehlednější.

Å Upozornění!

Parametr Command v akci Send Command musí být shodný s parametrem Command v globální dynamizaci OnCommand, aby se akce na serveru spustila.

8.2 Skript hromadného nastavení hodnoty

Hromadné nastavení hodnoty na základě příkazu OnCommand

Ve stromové struktuře projektu v dynamizacích některého modulu, nejlépe do modulu příslušné paměťové databáze, vložíme dynamizaci OnCommand.



Ve vlastnostech a dynamizacích nastavíme:

Typ podmínky:On CommandCommand:CMD_COPYTyp akce:Bara Script

🗾 Poznámka:

Dynamizace OnCommand se nespustí za žádných jiných okolností, než bezprostředně po odeslání stejného příkazu dynamizací Send Command.Nemůže tak dojít k náhodnému, či nechtěnému, provedení dynamizace, například po restartu systému.

Upravíme cílový skript tlačítkem Editovat. Následuje příklad skriptu pro bezpečné hromadné nastavení, z kontrolou zdrojové hodnoty pro nastavení a jejím resetováním po provedení.

```
program HromadneNastaveni;
procedure SetDifValue( ref dest : TIOChannelBinary; src : binary );
begin
       11
       // nenastavujeme do nevalidni promenne
       if dest.Valid = 0 then
              return;
       11
       // nenastavujeme shodnou hodnotu znovu
       if dest = src then
              return;
       11
       // samotne nastaveni cilove hodnoty
       dest := src;
end;
qlobal
       value : binary;
begin
       11
       // zdrojova hodnota ma hodnotu <Nezadano>, niz nenastavujeme, konec
       if (MD.SourceValue = 0) then
              return;
       11
```

```
// cilovou hodnotu budeme nastavovat na hodnotu FALSE (VYPNUTO)
if MD.SourceValue = 1 then
        value := false;
//
// cilovou hodnotu budeme nastavovat na hodnotu TRUE (ZAPNUTO)
if MD.SourceValue = 2 then
        value := true;

SetDifValue( MD.TargetValue1, value );
SetDifValue( MD.TargetValue2, value );
SetDifValue( MD.TargetValue3, value );
//
// nastaveni zdrojove hodnoty do bezpecneho stavu <Nezadano>
MD.SourceValue := 0;
end.
```

Ukázkový skript je přímo ve zdrojovém textu komentován. Principem je, že je ošetřeno, aby se cílové technologické hodnoty zadávaly jen jednou. Jsou uvedeny jako první parametr procedury SetDifValue, která kontroluje validitu cílové proměnné a různost její hodnoty od zadávané - druhého parametru. Před samotným spuštěním jednotlivých nastavení se kontroluje, zda zdrojová hodnota byla zadána uživatelem a připravuje s proměnná s požadovaným stavem pro nastavení do všech cílových technologických proměnných. Po provedení všech nastavení proměnných se okamžitě resetuje zdrojová proměnná do nezadaného stavu, aby nemohlo dojít k vícenásobnému náhodnému provedení tohoto přiřazovacího skriptu.

💢 Tip!

Pomocí příkazu #include "soubor.bal" můžeme na místo jednotlivých řádků vložit textově volání funkcí pro nastavení hodnot. Externí soubor připravíme v podadresáři Scripts projektu a bude obsahovat pouze právě tato volání SetDifValue(...) pro jednotlivé proměnné. Systém pak při provádění skriptu na místo #include obsah dočasně textově vloží.

9 Alarmy, dynamizace

Tato kapitola bude věnována popisu alarmů, alrmních stylů, dynamizací a příkladů s tímto tématem, zejména pak:

- popis a význam alarmních stylů
- zpoždění odesílání alarmů napříkal pro snížení počtu odesílaných SMS a e-mailů
- Hromadná, bezpečná nastavování hodnot s potvrzením a kontrolou
- Změna stylu alarmů, zpoždění alarmů
- Odeslání hodnot a alarmů emailem

🗾 Poznámka:

Kapitoly na sebe navazují, projděte si je postupně a zkuste si uvedené příklady sami zrealizovat.

9.1 Alarmní styly

Alarmy v Monitorovacím systému ProCop mají přiřazen globální, modulový, či uživatelský styl. Standardně je nejlepší používat globální styly, změna chování se pak provádí na jednom místě a projeví se ve všech místech použití.

Název	Popis	Тур	Priorita		Text	Bary	/a	s	Tisk	P S	Zpožděn	Text	Barva S 1	Tisk P	S Zpož
)														
🐊 Alarm	Styl pro kanálový alarm	Alarm	💌 Vysoká	•	Normál:		: x		1	V	0 s	Porucha:	(💌 📝	v	📃 0 s
🐊 Alarm Reset	styl pro reset alarmů	Alarm	Nízká	•	Reset OK:		i 💌			v	0 s	Reset nutný:	· 💌 📃	V	📃 0 s
🐊 Fault	Styl pro alarm bez pageru	Alarm	💌 Vysoká	•	Normál:				1		0 s	Porucha:	(💌 📝	V	📃 0 s
🕼 Fault Reset	Styl pro reset alarmů	Alarm	Nízká	•	Reset OK:		i 💌				0 s	Reset nutný:	· 💌 📃		🔲 0 s
🥼 Warning	Styl pro kanálovou výstrahu	Alarm	Normální	•	Normál:				1		0 s	Výstraha:		V	📃 0 s
🥼 Sensor Error	Styl alarmů poruch čidel	Alarm	Normální	•	Čidlo normál:		i 💌		1		0 s	Čidlo porucha:	i 💌 🔽	V	📃 0 s
🐊 Set Value	Styl nastavení hodnoty kanálu	Událost	💌 Vysoká	•	Nastaveno:		•		1		0 s	Porucha:	(💌 🗸	1	0 s
🥼 No Log Value	Styl pro vypnutí logování kanálů	Vypnuto	Nejnižší	•	Normál:		į v		\checkmark	\checkmark	0 s	Porucha:	(× 🗸	1	0 s
🥼 Comm Alarm	Styl pro komunikační alarm	Alarm	Nejvyšší	•	OK:		i 💌		1	V	0 s	Alarm:	· · ·	v	📃 0 s
🐊 Comm Event	Styl pro komunikační událost	Událost	💌 Vysoká	•	OK:		: <u>-</u>		1		0 s	Událost:	· 💌 📰	V	📃 0 s
🥼 Comm Log	Styl pro komunikační logování	Log	💌 Vysoká	•	Modul:		(💌				0 s	Porucha:	(× 🗸	\checkmark	0 s
🐊 Way Alarm	Styl alarmů z komunikačních cest	Alarm	Nejvyšší	•	OK:		5		1		0 s	Cesta:	F 💌 🔊	V	📃 0 s
🥼 Bara Script Alarm	Styl alarmů BáraSkriptů	Alarm	💌 Vysoká	•	OK:		i 💌		1		0 s	Bára:	i 💌 📝	V	🔲 0 s
🐊 System Alarm	Styl systémových alarmů	Alarm	Nejvyšší	•	OK:		5		1		0 s	SysAlarm:	í 💌 📝	V	📃 0 s
🐊 System Event	Styl systémových událostí	Událost	Nejvyšší	•	SysEvent:		i 💌		1		0 s	SysEvent:	· 💌 🕅	V	🔲 0 s
🐊 System Log	Styl systémových logování	Log	Nejvyšší	•	SysLog:				1		0 s	SysLog:	í 💌 📰	V	📃 0 s
🥼 ProSQL Alarm	Styl pro ProSQL alarm	Alarm	Nejvyšší	•	OK:		i 💌		1		0 s	Alarm:	· · ·	v	🔲 0 s
🥼 ProSQL Event	Styl pro ProSQL událost	Událost	💌 Vysoká	•	OK:		i 💌		1		0 s	Událost:	• 💌 🥅	V	📃 0 s
🥼 Note	Styl pro poznámky	Alarm	Normální	•	Zrušeno:		i 💌				0 s	Poznámka:	· 💌 🕅		🔲 0 s
🐊 Alarm Hidden	Styl pro kanálový alarm	Skrytý	Vysoká	•	Normál:				V	1	5 s	Porucha:	i 💌 🔽	v	5 s

Globální alarmní styly

Každý styl alarmu má společné vlastnosti a také vlastnosti pro úroveň 0 = zánik a 1 = vznik. Styl alarmů má tyto společné vlastnosti :

- Název používá se pro identifikaci stylu v systému
- Popis pouze informativní pro snazší orientaci a určení
- Typ přiřazuje jeden z následujících typů hlášení:
 - o Vypnuto hlášení se vůbec neodesílá, nezaznamenává
 - Alarm alarmní hlášení, červený zvoneček
 - Událost události, zelený zvoneček, obvykle nastavení hodnot, nebo logování komunikačních modulů
 - o Log systémový zápisník, modrý zvoneček, systémové události, zápisy stavy a poruchy
 - o Poznámka pouze poznámky k displejům, žlutý zvoneček (jen s alarmy v SQL)
 - o Sktytý nikde se nezobrazuje, slouží k přeposílání alarmů se zpožděním (jen s alarmy v SQL)
- Priorita pro filtr v prohlížeči alarmů

Pro úroveň zániku a vzniku se nastavují ve stylu následující vlastnosti :

- Text tento text se přidá pro na začátek alarmního textu pro danou úroveň
- Barva touto barvou se v prohlížeči alarmů bude zobrazovat text celého řádku daného hlášení pro danou úroveň
- Příznaky Ano/Ne pro rozlišení různých vlastností zpracování hlášení, mohou, ale nemusí být implementovány v projektu
 - Siréna pro hlášení (Alarmy) s tímto příznakem při dosažení dané úrovně se spouští zvuková signalizace
 - o Tisk hlášení s příznakem tisku mohou být zasílány k tisku
 - o Pager pro rozlišení hlášení, která je třeba odesílat jako SMS, E-mail, atp.
 - Speciál univerzální příznak, například pro druhý cíl SMS, e-mail, atp.
- Zpoždění časový úsek, kdy alarm čeká, zda se nevrátí do původní úrovně; pokud ano, vůbec nikde se neobjeví, v opačném případě se zahlásí s původním časem

🗾 Poznámka:

Zpoždění se definuje pro každou úroveň zvlášť, jelikož může být požadavek na různou hodnotu zpoždění pro vznik a zánik hlášení. Obvykle se však do obou časů uvádí shodná hodnota.

Uživatelsky definovaný globální alarmní styl

Při potřebě většího rozlišení alarmů například pro více cílů zobrazení, SMS a e-mailů, nebo při potřebě odesílání SMS, emailů a ž po nastavené době, kdy alarm trvá, definujeme uživatelské globální alarmní styly. Tyto obvykle zakládáme na již existujícím alarmním stylu, při vložení nového se systém zeptá, ze kterého stylu se bude vycházet.

Alarmům v komunikačních modulech pak přiřazujeme styly tak, abychom je dle potřeby dalšího zpracování a filtrování rozlišili, například podle jména stylu, nebo některého z příznaků.

Modulové a uživatelské alarmní styl

Pokud nechceme, aby různé moduly používaly stejné alarmní styly, můžeme definovat alarmní styly přímo u každého modulu. Takto definované styly lze použít jen u alarmů, které jsou definovány u tohoto modulu. Další možností je výchozí alarmní styl modifikovat přímo u každé proměnné. Takováto úprava však má za následek, že jakoukoliv změnu je nutné udělat přímo u konkrétní proměnné.

襘 Tip!

Používejte co nejvíce globální, nebo modulové alarmní styly. Uživatelská modifikace stylů u konkrétních proměnných může vést ke zvýšení pracnosti při údržbě a budoucích změnách v projektu.

💢 Tip!

V přehledu alarmů u každého module je vidět veškeré použité styly jednotlivých alarmů a lze je hromadně upravovat.

9.2 Dynamizace OnAlarm

Dynamizace **OnAlarm** se provádí na základě vzniku jakéhokoliv hlášení v systému. Podle nastavení filtrů ve vlastnostech a dynamizacích se rozhodne o provedení dynamizační akce. Dynamizaci OnAlarm lze vložit v datovém serveru u libovolného modulu do dynamizací, nebo ve vizuálním projektu do vizuálních dynamizací. **Příklady použití dynamizace OnAlarm dle umístění:**

• v dynamizacích modulu, v datovém serveru

- o nastavení technologické hodnoty jako reakce na alarm
- o přeposlání alarmu s jiným stylem, pro zpoždění odeslání SMS, e-mailu
- o odeslání SMS, e-mailu
- ve vizuálních dynamizacích, na klientovi ve vizuálním projektu
 - o spuštění sirény, zobrazení hlášení v oznamovací oblasti
 - o přepnutí obrazovky na základě alarmu

🍕 Vlastnosti a dynam	nizace	Ð
	Hodnota	
💷 Název dynamizace	Ukázka OnAlarm	
💷 Typ podmínky	On Alarm	
🖃 Parametry podmínk	Y	
🔲 Zakázáno	Ne	
🔲 Systém		
🔲 Zdroj		
💷 Zařízení		
🔲 Styl	Alarm)
🔲 Alarm	📝 <ano></ano>	
💷 Událost	📃 <ne></ne>	
💷 Logování	📃 <ne></ne>	
🔲 Skrytý	📃 <ne></ne>	
🗆 Úrovně		
💷 Pro úrovně:	Všechny	
🔲 Úroveň (0=normá	<mark>lj</mark> O	
🗆 Priority		
💷 Pro priority:	Všechny	
🔲 Priorita	Normální	
🔲 Siréna	Cokoliv	
🔲 Tisk	Cokoliv	
🔲 Pager	Cokoliv	
🔲 Speciál	Cokoliv	
🗉 Typ akce	Bara Script	
Provést pokud 0 <> pro	han	***
💷 Zpoždění	0 s	
Parametry akce		
💷 Název skriptu		
🔲 Maximální doba běhu	500 ms	
🗆 Příznaky		
Řetězcové operac	e 🔲 <ne></ne>	
🔲 Zásobník parametrů [256	
🔲 Max. alokovaných ře	<mark>t</mark> 64	
BaraScript	Editovat!	
Zakázán běh nebo la	d	
Kompilovat	Kompilovat!	
Výsledek kompilace		

Ukázka dynamizace OnAlarm

Zakázáno

Dynamizaci lze zcela jednoduše zakázat, aby se neprováděl

Systém, zdroj, zařízení

Pokud chceme filtrovat zachycený alarm například podle zdroje, napíšeme do parametru **Zdroj** jeho název. Stejně to platí i pro **Systém a Zařízení**. Tyto položky se dají kombinovat. Pokud všechny tři položky ponecháme prázdné, na Systém, Zdroj i Zařízení alarmu může být libovolný.

Å Důležité upozornění!

Pokud vyplníme nějaký z parametrů systém, zdroj, zařízení, musíme zbylé parametry vyplnit znakem hvězdička '*' (zástupný symbol s významem "všechny"), pokud nemusí být ostatní parametry alarmu skutečně prázdné.

Styl

Filtrování reakce na alarm můžeme provést také podle přesného názvu stylu. Při prázdném zadání může být název stylu libovolný.

Tipy a triky

Alarm, událost, logování, skrytý

Zaškrtávací políčka alarm, událost, logování a skrytý slouží pro další filtrování zachyceného alarmu dle typu. Pokud zatrhneme více položek současně, akce se provede, pokud je typ alarmů kterýkoliv ze zatržených.

🗾 Poznámka:

Pokud nebude zaškrtnuté žádné políčko dynamizační akce se nikdy neprovede, filtru nevyhoví žádné hlášení.

Úrovně

Provedení dynamizační akce můžeme navíc podmínit i úrovní hlášení, zda alarm vzniká (1) nebo zaniká (0).

Priority

Taktéž lze odfiltrovat alarmy dle priority.

Priority

Častým rozlišení hlášení pro provedení akce bývá některý z příznaků alarmního stylu, jako je Siréna, Tisk, Pager a Speciál. Jednotlivé kombinace musí být takové, jaké se nastaví v parametrech podmínky. Volba "Cokoliv" vypovídá o svém významu již názvem.

Pokud jsou výše uvedené podmínky splněny současně (AND) a až na položky typ, kde jsou zatrhávací pole ve vzájemném významu (OR), je nastavená dynamizační akce provedena.

🚖 Tip!

Projekt s příkladem dynamizace OnAlarm naleznete v sekci ke stažení, ProCop 3.7 pod názvem Tips and Tricks Sample 1.

9.3 Zpoždění odesílání alarmů

V reálném projektu se často stává, že některá technologická hodnota zakmitá, způsobí alarm, jeho zánik, na krátkou chvíli a to i několikrát po sobě, ač se nejedná významný problém. Příkladem může být dopouštění nádrží, tlakování systémů, rušení na vedení, atp.

Monitorovací systém ihned při zjištění alarmního stavu ten zaznamená a reaguje dle nastavení (oznámením, sirénou, atp.). Pokud však alarm v následujících vteřinách opět zanikne, reaguje systém opět na jeho zánik, jak je nastaven. **Pro potlačení tohoto jevu je možné provést následující kroky:**

- nastavit lépe technologii, upravit program PLC, aby k takovému jevu nedocházelo, byl potlačen již na úrovni regulace - nejlepší řešení technologických zákmitů
- příslušnému alarmnímu sty lu 67 nastavit krátké zpoždění pro vznik a zánik pokud alarm trvá kratší, než nastavenou dobu, nebude vůbec nikde zaznamenán - možné řešení, zcela se potlačí tento jev, uživatel není nijak informován
- přeposláním alarmů s jiným stylem se zpožděním původní alarm se vyhlásí/zanikne okamžitě, alarm s jiným stylem se vyhlásí, jen pokud stav trvá déle, než nastavená doba pak se reaguje např. sirénou, SM S, e-mailem

Použití přeposlání alarmů s novým stylem

Pokud chceme původní alarm zachovat, ale uživatele neinformovat, například sirénou, SMS, či e-mailem 96 dříve, než tento alarm potrvá alespoň nastavenou dobu, pak provedeme následující:

- původní styl alarmu upravíme tak, aby neměl zpoždění a měl vypnuty příslušné příznaky, na základě kterých se spouští siréna, posílá SMS, e-mail 961, atp.
- vytvoříme nový alarmní styl, na základě původního; pokud nechceme tento podruhé zobrazovat v prohlížeči alarmů, nastavíme jeho typ na "Skrytý" 67
- novému stylu nastavíme požadované zpoždění a vlastnosti, které má narozdíl od původního přenášet siréna, pager, atp.
- v dynamizacích datového serveru vytvoříme dynamizaci OnAlarm 68 pro všechny alarmy původního stylu s dynamizační akcí ResendAlarm, kde pouze změníme styl na nově vytvořený
 - tímto postupem se původní alarm zapíše okamžitě do databáze, a přepošlete se s novým stylem, se zpožděním
- v dynamizacích datového serveru vytvoříme druhou dynamizaci OnAlarm 68 pro všechny alarmy nového stylu a provedeme odeslání SMS, nebo e-mailu dynamizací SetValue a přířazením do cílové proměnné výsledku funkce CmdLineAlarm()
 - toto provádíme, pokud na základě zpožděného alarmu chceme odeslat SM S/e-mail se zpožděním, pokud alarm trvá přednastavenou dobu

v dynamizacích vizuálního projektu zkontrolujeme, že existuje, nebo upravíme či vytvoříme dynamizaci
 OnAlarm 68, s příznakem syrény, typem "Skrytý" a akcí přehrání zvuku

対 Tip!

Projekt s příkladem dynamizace OnAlarm naleznete v sekci ke stažení, ProCop 3.7 pod názvem Tips and Tricks Sample 1.
10 Moduly a proměnné

V této části budeme v podkapitolách postupně dokumentovat některé moduly monitorovacího systému ProCop z pohledu tvůrce projektu, jejich vlastnosti a použití. Společné vlastnosti proměnných popisovaných modulů, které mohou (a nemusí) proměnné mít popisuje společně následující odstavec.

Vložení modulu

V Panelu nástrojů ProCop otevřeme v režimu Designer Správu projektu, vybereme Tento počítač\Cesty a moduly vložíme tlačítkem Ins, nebo symbolem plus v nástrojové liště modul. Moduly, které nekomunikují, nebo nepoužívají komunikační cesty se vkládájí přímo do kořene, jako například:

- Memory Database
- Global Modul
- Control M odul
- Signal Generator
- a mnohé další

Komunikující moduly musí být vloženy na příslušnou komunikační cestu, případně pod konkrétní komunikační protokol.

Každý modul musí bát pojmenován, v rámci daného projektu, nebo koncentrátoru unikátně.

M odul musí mít definován unikátní prefix proměnných v rámci projektu. Tento je pak součástí názvu každé proměnné a je od něj oddělen tečkou.

Popis společných vlastností proměnné modulů monitorovacího systému ProCop

Všechny moduly systému ProCop mají ve stromu položku Proměnné a také podskupiny systémových proměnných pro ovládání modulu System. U většiny modulů můžeme vytvářet uživatelsky libovolné skupiny pro členění a organizaci proměnných.

Vkládání skupin a proměnných provádíme tlačítkem Ins, nebo symbolem plus v nástrojové liště.

Na příkladu uveď me základní společné vlastnosti proměnných, které dle konkrétních modulů mohou, ale nemusí být k dispozici. Příklad je uveden na proměnnou typu Discrete, s uživatelsky definovanou tabulkou významů hodnot:

🖏 Vlastnosti a dynamizace 📮 🛙					
	Hodnota				
🖩 Název	MD.SourceValue				
🖩 Тур	Discrete	•			
🗉 Popis	Zdrojová hodnota				
🗉 Pole	1				
🗉 Použito	Ano>				
Inicializace	0				
🗉 Ukládat poslední hodnotu	Ne	•			
🔲 Ukládat do inicializace (pio)	Ne	$\overline{\mathbf{v}}$			
🗆 Limity, přepočty					
🗉 Dolní mez	0				
💷 Horní mez	2				
Násobitel	1				
🗉 Posun	0				
💷 Jednotka		-			
Formát výpisu	<user table=""></user>				
🗉 Tabulka textů	0: <nezadáno> 1:Vypnuto 2:Zapnuto</nezadáno>	•••			
Styl záznamu	Set Value				
🗄 Alarm					
🗄 Trend					

Vlastnosti a dynamizaci proměnné v modulu Memory Database

Název

Název proměnné, zobrazuje se včetně prefixu modulu. Toto celé jméno smí obsahovat jen alfanumerické znaky a musí být unikátní v rámci celého monitorovacího projektu.

Тур

Datový typ proměnné, Analog, Binary, Counter, Discrete, nebo Text.

Popis

Libovolný textový popis proměnné, vypisuje se při vzniku a zániku případného alarmu, definovaného u této proměnné a jako implicitní hodnota názvu trendu.

Pole

Standardně obsahuje jedna proměnná jednu hodnotu. Pokud by proměnná měla obsahovat více hodnot, napíšeme do tohoto parametru jejich počet. Na jednotlivé prvky pole přistupujeme pomocí operátoru hranatých závorek: NázevProměnné[x]

Použito

Tato položka se obvykle nezadává ručně. Slouží k vyhodnocení, zda se někde v systému používá. Tento stav se nastavuje jen při překladu celého projektu (modulů, dynamizací, displejů), resetovat se musí ručně.

Inicializace

Výchozí hodnota proměnné při její inicializaci - spuštění projektu. Pokud je nastaven příznak "Ukládat poslední hodnotu", pak se systém při spuštění pokusí vyčíst hodnotu proměnné z dočasných souborů, pokud není nastaveni příznak "Ukládat do inicializace (pio)".

Horní a dolní mez

Slouží pro nastavení hodnoty, zobrazuje se v nastavovacím dialogu a ten neumožní nastavit uživateli hodnotu mimo zadané meze. Dále může sloužit pro definici mezí připojeného alarmu.

Násobitel a posun

Jedná se o standardní přepočet hodnoty kdy $y = k^*x + q$, kde:

- y hodnota proměnné v systému
- x původní nastavená hodnota do proměnné
- k násobitel (zadávaná vlastnost proměnné)
- q posun (zadávaná vlastnost proměnné)

Formát výpisu / <User Table>

Popisuje formát, kterým bude daná proměnná vypsána na displej. Každý typ proměnné má několik přednastavených formátů výpisu. Pokud proměnná typu discrete reprezentuje specifický konečný počet stavů nebo u proměnné typu binary chceme přejmenovat jeden ze dvou stavů, zvolíme volbu **Tabulka textů**>. U nově zobrazeného parametru **Tabulka textů** u kterého klikneme na tři tečky. V zobrazené tabulce si pak výpis formátu vytvoříme podle sebe. V obrázku jsou uvedeny tři stavy. Těchto stavů samozřejmě může být libovolný počet. Hodnota se nesmí v tabulce opakovat.



User Table ve Formátu výpisu

Styl záznamu

Styl pro událost, pokud dojde k nastavení hodnoty. Implicitně je přednastaven styl Set Value, kde je nastavení v systému zaznamenáno jako událost a vypisuje se zeleně. Pokud nechceme nastavení zaznamenávat, zvolím styl No Log Value.'

10.1 Uživatelský formát proměnných

Formát výpisu binárních hodnot je od verze ProCop 3.3 nastavitelný obdobně, jako u diskrétních hodnot. V tabulce vlastností a dynamizací se nově zobrazuje náhled jednotlivých významů hodnot, zadaných v tabulce.

Uživatelská tabulka a význam hodnot u diskrétních (binárních) kanálů

Vlastnosti a dynamizace		X
	Hodnota	
💷 Název	R1.Ahu02OpModMan	
🗉 Тур	Discrete	
Popis	Manuálně mód VZT2	
🔲 Použito	✓ <ano></ano>	
Resetovat!	Ne>	
🗄 BACnet objekt		
🗆 Limity, přepočty		
💷 Dolní mez	1	
💷 Horní mez	4	
Násobitel	1	
Posun	0	
💷 Jednotka		
Formát výpisu	<user table=""></user>	•
🗉 Tabulka textů	1:Auto 2:Vypnuto 3:Tlumený 4:Plný	***
🖬 Styl obnovy	<module default=""></module>	•
💷 Styl záznamu	Set Value	•
🗄 Alarm		
🕀 Trend		

Uživatelské formáty binárních a diskrétních hodnot

Pokud přednastavené hodnoty nevyhovují potřebám projektu, stačí vybrat volbu <User Table> a nastavit si v tabulce významy jednotlivých hodnot podle potřeby. V tabulce se pro diskrétní hodnoty dají zadat hodnoty 0 - 255 a k nim příslušné texty. U binárních hodnot mají význam jen hodnoty 0 a 1. Ostatní jsou ignorovány.

Uživatelská tabulka a význam hodnot u diskrétních (binárních) kanálů

Т.	abulka textů	
	🐰 💼 📓 💠 🂠 💥 🏘 🧖	
Hodn	Text	
1	Auto	
2	Vypnuto	
3	Tlumený	
4	Plný	
		🧹 Zavřít

Kopírování uživatelských formátů

Kopírování můžeme provádět v tabulce označením a kopírováním řádků. Nově je možné přímo v tabulce vlastností a dynamizací vybrat řádek Tabulka textů a obvyklou kombinací kláves Ctrl+C formát zkopírovat a následně vložit do Tabulky textů jiné diskrétní, nebo binární proměnné kombinací Ctrl+V.

10.1.1 Nastavovací dialogy

Pro diskrétní kanály, které mají nastaven formát výpisu na uživatelskou tabulku, nebo přednastavený výčet a pro všechny binární kanály, jsou od ProCop verze 3.3 zobrazovány výběry v nastavovacích dialozích. Nastavení hodnoty binárního kanálu - nyní jen výběr hodnoty

Nastavení hodnoty						
VZT4: Povolení alarmů mezí ⊙ Zapnuto ⊚ Vypnuto	✓ <u>Q</u> K ★ Storno ? Nápověda					

Dialog pro nastavení hodnoty (SetValueDialog) je pro diskrétní a binární hodnoty automaticky doplněn přepínači se všemi popsanými hodnotami formátu. Meze a zadávací řádek je zrušen.

Nastavení hodnoty diskrétního kanálu - automaticky vygenerované přepínače

Nastavení hodnoty							
Režim provozu VZT4 ⊘ Automat ⊚ Vypnuto ⊚ Tlumený provoz ⊘ Plný provoz	<u>Q</u> K <u>S</u> torno <u>N</u> ápověda						

V nastavovací tabulce hodnot jsou k dispozici vždy jen výběry hodnot pro binární a pro diskrétní hodnoty s uživatelskou tabulkou hodnot. Formáty se nastavují přímo u kanálů 75 a není potřeba je předefinovávat v displeji u konkrétního dialogového okna.

Nastavení hodnoty v tabulce - diskrétní a binární kanál s uživatelským formátem

Tabulka hodnot		
VZT4: Režim Filtr přívod 54.4.2	Tlumený ▼ Normal ▼	<u> </u>

10.2 Memory Database

Monitorovací systém ProCop disponuje modulem pro dočasné, nebo trvalé uložení a zpracování hodnot. Tímto modulem je Memory Database - modul paměťové databáze.

K čemu lze modul Memory Database použít:

- dočasnému uložení hodnoty, po restartu monitorování se inicializuje na přenastavenou hodnotu
- k uložení nedůležité hodnoty, která se často mění, po restartu monitorování se poslední hodnota načte z dočasných datových souborů projektu (.IOD) - nearchivuje se
- k trvalému uložení důležité hodnoty hodnota se po změně uloží do inicializace v projektovém souboru modulu (.PIO) - archivuje se
- k přepočtu hodnoty, pomocí dynamizace
- k definici alarmních stavů
- pro trendování hodnoty z výpočtu

Proměnné modulu lze pojmenovat podle potřeby, členit libovolně do skupin, definovat k proměnným iniciální hodnotu a způsob ukládání, meze, přepočty, jednotky, formát výpisu a také trendy a alarmy.

Popis paměťové proměnné a ukládání její hodnoty

V Panelu nástrojů ProCop otevřeme v režimu Designer Správu projektu, vybereme Cesty a moduly vložíme modul **Memory Database**.

V položce Proměnné právě vytvořeného modulu vytvoříme novou proměnnou. Nastavení proměnné probíhá v jejích vlastnostech a dynamizacích. Příklad je uveden na proměnnou typu Discrete:

🍕 Vlastnosti a dynamizace					
	Hodnota				
Název	MD.SourceValue				
🗉 Тур	Discrete	•			
Popis	Zdrojová hodnota				
🗉 Pole	1				
🗉 Použito	Ano>				
Inicializace	0				
🗉 Ukládat poslední hodnotu	Ne				
Ukládat do inicializace (pio)	Ne	v			
🗆 Limity, přepočty					
💷 Dolní mez	0				
🔲 Horní mez	2				
Násobitel	1				
🖪 Posun	0				
💷 Jednotka					
Formát výpisu	<user table=""></user>				
🗉 Tabulka textů	0: <nezadáno> 1:Vypnuto 2:Zapnuto</nezadáno>	•••			
Styl záznamu	Set Value				
🕀 Alarm					
🗄 Trend					

Vlastnosti a dynamizaci proměnné v modulu Memory Database

Základní vlastnosti proměnné jsou popsány v předchozí kapitole 731.

Inicializace

Výchozí hodnota proměnné při její inicializaci - spuštění projektu. Pokud je nastaven příznak "Ukládat poslední hodnotu", pak se systém při spuštění pokusí vyčíst hodnotu proměnné z dočasných souborů, pokud není nastaveni příznak "Ukládat do inicializace (pio)".

Ukládat poslední hodnotu

Tuto volbu zvolíme, pokud chceme mít přístupnou hodnotu proměnné i po zastavení a opětovného spuštění projektu,

restartu počítače. Po spuštění projektu se nejprve hodnota načte z inicializace (z PIO). Pokud není nastavena volba "Ukládat do incializace (pio)" je hodnota vyčtena z dočasného binárního souboru, pokud se toto podaří a nastavena na hodnotu při posledním korektním vypnutí monitorování. Ukládání probíhá do dočasného souboru, který není součástí archivace projektu, může být kdykoliv smazán uživatelem, poslední stav nemusí být uložen, nebo soubor může být při zápisu poškozen, např. tvrdým resetem počítače. Tyto soubory nemusí být správně načteny ani po přeinstalaci systému ProCop na novější verzi.

💢 Tip!

Tuto volbu preferujeme, pokud se hodnota často mění, například periodickým skriptem, od komunikace a podobně a také pokud může být kdykoliv při spuštění systému nastavena na inicializační hodnotu.

Ukládat do inicializace (pio)

Tato možnost nám zajistí, že o hodnotu proměnné nepřijdeme ani při výše zmíněné ztrátě dočasného souboru. Hodnota se v krátké době po změně zapíše přímo do inicializační hodnoty ve zdrojovém projektovém souboru modulu (PIO). Nastavení hodnoty má za následek změnu hodnoty ve vlastnosti "Inicializace". Při archivaci projektu je tato hodnota spolu s modulem zálohována.

💢 Tip!

Tato volba je bezpečná pro ukládání hodnoty, je však systémově náročná. Není vhodné ji používat pro často se měnící hodnoty, například skriptem.

10.3 Ukládání stavových proměnných

V případě požadavku na ukládání stavových proměnných systému jako je například datum pro export dat zadávané obsluhou je možno využít IO modul Memory Database (77) (paměťová databáze). Tento modul inicializuje hodnoty na hodnotu 0, pokud není zadáno konfiguračně jinak.

Proměnné jsou implicitně ukládány do souborů *.iod, které jsou považovány za dočasné a jejich formát a obsah může být mezi verzemi systému ProCop změněn. Také může dojít k destrukci jejich obsahu například po výpadku napájení, havárii disku apod.

Å Důležité upozornění!

Nedoporučujeme ukládat žádná kritická data do těchto proměnných, při výpadku napájení, nekorektním ukončení aplikace, nebo upgradu systému může dojít k jejich ztrátě!

Pokud je zapotřebí vyšší bezpečnost pro uložení hodnot stavových proměnných, doporučujeme použít volbu "Ukládat do inicializace (pio)". V tomto případě dojde k přepsání inicializační hodnoty ve zdrojovém souboru projektu typu *.pio a mělo by následně dojít k archivaci projektu obsluhou tak, aby nemohlo dojít k přepsání stavové proměnné při obnovení projektu ze zálohy (v případě pádu disku apod.).

Pokud je zapotřebí ukládat stavové proměnné s nejvyšší úrovní bezpečnosti, doporučujeme využít externí úložiště jako např. SQL server případně CSV, DBF či INI soubor například na externím zabezpečeném disku. Při využívání stavových proměnných z externích úložišť vysoce doporučujeme kontrolovat stav a validitu dat získaných z těchto úložišť.

🞽 Poznámka

Nastavením vlastnosti "Ukládat do inicializace (pio)" zvýšíte bezpečnost přenosu jejich hodnoty mezi starty systému i při upgrade. Nevyužívejte však pro často se měnící proměnné (skriptem), aby nedocházelo k neustálým modifikacím zdrojového souboru. Vždy je nutné si uvědomit, že pokud i tak dojde ke ztrátě hodnoty, iniciální "tovární" hodnota je 0. Tento stav by měl být pro následné využití považován za bezpečný a s touto logikou je nutné v systému uvažovat.

10.4 Parametry modulu PX

Parametry komunikace modulu PX jsou obdobné jako u ostatních komunikačních modulů. Výjimkou je chování objektů schopných zasílat změny COV (ChangeOfValue).

79

	Hodnota	
🔲 Název	PZ	
🔲 Prefix	PZ	
🖬 Popis		
Soubor	pz.pio	•••
🔲 Zakázáno	Ne	•
🗄 O modulu	PxModule	
🗆 Parametry komunikace		
🖬 Adresa (hexa)	00200401	
Permanentní obnova	1h	
💷 Počet opakování	3	
💷 Styl alarmu komunikace	<comm alarm=""></comm>	•
🗆 Perioda obnovy		
🖾 Perioda (0=vyp)	5 m	
💷 Posunutí periody	Os	
💷 Obnov při startu	🔲 <ne></ne>	
Validita (0=vždy)	Os	
💷 První obnova (0=od půlnoci)	Os	
💷 Poslední obnova (1d=do půlnoci)	1 d	
🗆 Parametry regulátoru		
Typ regulátoru	PXC100	•
Verze PXC	Desigo 4.x	×
💷 Priorita ovládání	8	
COV perioda (0=vyp)	10 m	
💷 Hlášení poruch čidel	Sensor Error	•
🗆 Speciální funkce		
Soubor autokonfigurace	Bagetarna_Olomouc_E	•••
🔲 Opravit adresy pinû v CSV	📝 <ano></ano>	
💷 Název stanice	AS01	
💷 Ignorovat prefixy do úrovně	1	
🖬 Autogenerace	Autokonfigurace	
🔲 Rozdíl CSV proti autokonfiguraci	Autodiference	
🗉 Konverze textů ve skupinách		•••
🗉 Konverze textů při autokonfiguraci		•••
🗆 Podrobná nastavení		
Kontrolovat vždy po startu	Ne	•
🔲 Chybu čtení jako událost	Ne	•
Císt verzi databáze regulátoru	Ano	•
Převést název objektu na instanci	Ano	•
Kontrolovat název objektu a instanci	Ne	•
💷 Opravit chybu akceptace alarmu	Ne	•
🔲 Kontroluj verzi BACnet	Ano	•
Cti vlastnost AckRequired	Ne	•
🖬 Ignoruj BACnet Error rámce	Ne	•
💷 Opravuj adresy Pinů za běhu	Ne	•
🖬 Generuj alarm pro resetování	Ano	•
🔲 Generuj půlnoc v Scheduleru	Ano	•
Manuální režim v Scheduleru	Ano	
Cti výjimky ze Scheduleru	Ano	•
🔲 Ukaž seznam poruch v CommonEault		
	Ano	
Ukaž ComnAlarms v CommonFault	Ano Ne	

Vlastnosti modulu PX

10.4.1 Parametry regulátoru

Typ regulátoru

Nastavuje přesný typ regulátoru Desigo PX. Je nutno, aby bylo shodné se skutečným typem regulátoru a kontroluje se za běhu systému. V případě, že typ nebude zadán správně, komunikace se ukončí u bude hlášena porucha komunikace.

Verze PXC

Určuje, zda je firmware regulátoru ve verzi 2.3x, 4.0 nebo 5.0 (BACnet verze 1.0, 1.5 a 1.10). Kontrola je prováděna v případě zapnuté podrobné volby "Kontroluj verzi BACnet".

Priorita ovládání

Slouží pro prioritu při ovládání v BACnet vlastnosti PRIORITY_ARRAY. Po kompatibilitu s Desigo PXC ponechte na implicitní hodnotě 8.

COV perioda

Protokol BACnet je obecně schopen zasílat změny hodnot z regulátoru do nadřízeného systému (PC). Tuto schopnost podporují i podstanice řady Desigo PX.

Modul při komunikaci požádá regulátor o zasílání změn dat k jednotlivým proměnným s maximální délkou trvání implicitně 30 minut. Po vypršení této doby pak modul vyčte všechna data a následně opět požádá regulátor o registraci a zasílání změn. V periodickém čase obnovy pak modul pouze čte čas regulátoru případně čte proměnné, které mají zakázáno čtení COV nebo regulátor čtení přes COV nepodporuje. Modul u proměnných, kde COV probíhá pak zvyšuje čas obnovy v okamžiku příjmu změny či v době periodické obnovy (simuluje tak příjem dat).

🕭 Důležité upozornění!

V případě přetížení komunikační linky k regulátoru může být změna zaslaná regulátorem ztracena - regulátor se snaží provést doručení max. 3x. Z toho důvodu je nevhodné periodu COV příliš zvyšovat.

Vypnutí této možnosti lze provést nastavením hodnoty na 0. V tomto případě pak modul čte periodicky data v nastavené periodě modulu respektive v periodách daných stylem obnovy u proměnné (zcela standardním způsobem jako u jiných modulů s komunikací typu dotaz-odpověď).

Potlačení této schopnosti lze provést i u jednotlivých proměnných použitím vestavěného stylu obnovy "NoCOVobjects" případně tvorbou vlastního uživatelského stylu s vypnutou volbou "Používat COV".

🗾 Poznámka

Tato volba je k dispozici pouze pro modul v PC nikoliv v AlfaBox.

Hlášení poruch čidel

Styl alarmů pro hlášení poruch čidel. Porucha je indikována čtením BACnet vlastnosti Reliability.

10.4.2 Speciální funkce

Soubor autokonfigurace

Soubor autokonfigurace může být typu CSV nebo XLS a je generován pomocí nástroje EDE Export (v Desiogo 4.x a 5.x pak volbou nabídky). Soubor poskytuje seznam BACnet objektů, jejich typů popisů a dalších parametrů. V případě použití verze XLS je systém schopen jej přečíst pomocí rozhraní OLE Automation. V tomto případě je nutná instalace MS Excel. Systém pak soubor uloží do rozšířeného souboru CSV, který je možno číst i bez instalovaného systému MS Excel.

Výhoda použití EDE Exportu do XLS spočívá v tom, že v tomto formátu jsou k dispozici další informace oproti exportu do CSV - například texty popisující hodnoty diskrétních stavů. Toto platí pro Desigo XWorks do verze 4.x.

b Důležité upozornění!

Nástroj EDE Export verze 4.x nepracuje v některých případech korektně v případě exportu několika podstanic současně do jednoho souboru. Doporučujeme proto provést export pro jednotlivé podstanice do oddělených souborů typu XLS.

Od verze 5.0 jsou programem EDE Export generovány další 3 dodatečné soubory s názvy stejnými jako hlavní soubor rozšířené o přípony -Object-Types, -State-Texts a -Unit-Texts. V tomto případě je funkčnost v případě využití XLS a sady CSV souborů shodná.

Å Důležité upozornění!

Systém ProCop potřebuje všechny údaje o BACnet databázi v jednom souboru. Dodatečné soubory jsou proto automaticky importovány do hlavního souboru .CSV a dodatečným souborům je přidána nová přípona .MERGED. Kopírujte proto vždy do adresáře modulů všechny soubory (hlavní i dodatečné) společně a najednou !!!

Opravit adresy pinů v CSV

Volba způsobí opravu chybných adres pinů objektů generovaných EDE Exportem verze 5.0. Adresy budou opraveny při generování souboru CSV z XLS, při spojování souborů s popisy objektů a stupňů nebo při prosté autogeneraci či autodiferenci. Výsledný soubor CSV je použitelný jako zdroj pro systém ProCop starších verzí.

Seznam objektů, u nichž dochází k opravě je uložen v souboru PxOptions.xml, který je uložen v podadresáři Options instalace systému ProCop. Seznam je uložen v sekci "BadPinFunctions" a je možné jej ručně rozšiřovat. Standardně je dodáván soubor s tímto seznamem opravovaných objektů:

```
<BadPinFunctions>
       <Item0>
              <Function t="s">DelOds</Function>
       </Item0>
       <Item1>
              <Function t="s">DelPor</Function>
       </Item1>
       <Item2>
              <Function t="s">Hra</Function>
       </Item2>
       <Item3>
              <Function t="s">KickFnct</Function>
       </Item3>
       <Item4>
              <Function t="s">KickFnct1</Function>
       </Item4>
       <Item5>
              <Function t="s">KickFnct2</Function>
       </Ttem5>
</BadPinFunctions>
```

Název stanice

Musí zde být zadán přesný název stanice podle jména v EDE exportu. V opačném případě nebudou žádné proměnné importovány.

Ignorovat prefixy do hloubky

Umožňuje při tvorbě stromu proměnných ignorovat prvých N úrovní stromu. V případě zbytečně příliš hlubokého stromu proměnných zkuste zvýšit úroveň.

Autogenerace

Načte data z EDE exportu a vygeneruje z něj databázi regulátoru do podskupiny "Auto".

Rozdíl CSV proti autokonfiguraci

Volba Autodiference je vhodná v případě, že potřebuje najít větší počet změn po nahrání nového aplikačního softwaru do podstanice. V poměrně rozsáhlých stromech proměnných je poměrně obtížné najít, co aplikační programátor v podstanici doplnil, ubral či změnil.

💢 Tip!

Pokud je databáze "vyčištěná" od nepotřebných proměnných, doporučujeme před volbou Autodiference provést volbu Autokonfigurace s původní verzí CSV souboru, která zpět doplní zrušené proměnné. Volba Autodiference vygeneruje tři skupiny obsahující změny nové verze softwaru podstanice oproti současné databázi. Před volbou Autodiference je nutné nakopírovat nový soubor vytvořený pomocí nástroje EDE Export. Vytvoří se dočasné skupiny:

- "Removed channels, not use !" skupina obsahuje proměnné, které v nové verzi EDE exportu chybí
- "Added channels, not use !" obdobně obsahuje proměnné, které v nové verzi EDE Exportu přibyly
- "Dif. object type, not use !" skupina obsahuje proměnné, jejichž BACnet typ byl oproti stávající databázi změněn

Skupiny slouží pouze pro orientaci, tak aby bylo zřejmé, co se mezi verzemi software podstanice změnilo, nikoliv pro vizualizaci. V případě spuštění modulu jsou skupiny automaticky smazány.

Konverze textů ve skupinách

V případě potřeby náhrady standardních textů importovaných z EDE Exportu je k dispozici speciální nástroj, který dokáže generovat dle skupiny a standardního názvu objektu popisy proměnných. Slovník dokáže vkládat zvláštní text namísto zástupného symbolu @ a dále je schopen generovat definici alarmu, trendu a trendy řadit do skupin. Namísto zástupného symbolu je tak možno dynamicky vkládat dle potřeby například krátké názvy jako je TUV3 apod. Seznam a definice konverzního předpisu je možno dle potřeby upravovat či doplňovat.

対 Tip!

Konverzi textů ve skupinách je nutno u každé skupiny v jejich vlastnostech spustit stiskem tlačítka Konverze textů -Spustit. Před spuštěním je vhodné doplnit hodnotu zástupného symbolu @.

Konverze textů při autokonfiguraci

Systém Desigo PX nepoužívá v národní verzi háčky a čárky. Proto má modul implementován jednoduchý překladový slovník, který jednotlivé "ceske" výrazy při importu mění na "české". Slovník je možno doplňovat o vlastní konverzní páry textů dle potřeby.

10.4.3 Podrobná nastavení

Kontrolovat vždy po startu

Při každém startu monitorování provede kontrolu všech BACnet objektů, jejich typů, instancí (ignoruje uložené hodnoty od minule). Zjistí tak, pokud v PXC někdo od minulého spuštění přehrál SW. Kontrolu je možno vynutit v řízení modulů vyřazením a zařazením do obnovy.

Chybu čtení jako událost

Pokud nelze přečíst proměnná, generuje událost "Nelze obnovit proměnnou ..." jinak ji pouze znevalidní (do další pravidelné obnovy dat).

Číst verzi databáze regulátoru

Regulátor PX má uloženo pořadové číslo své databáze, které se po přehrátí SW automaticky zvyšuje. Po zvýšení tohoto pořadového čísla (on-line přehrátí programátorem za provozu dispečinku) modul automaticky provede kontrolu typů, instancí BACnet objektů apod.

Převést název objektu na instanci

Nutno nechat nastaveno u regulátorů, které si číslo instance při každé úpravě SW libovolně upraví a mění (např. řada Desigo PX).

Kontrolovat název objektu a instanci

V případě vypnutí předchozí volby (u regulátorů, kde je to pevně nastaveno), tak je to alespoň vhodné kontrolovat typ (ochrana proti chybě zadání databáze).

Opravit chybu akceptace alarmu

Potlačení chyby implementace kvitování alarmů v regulátorech Delta Controls.

Opravuj adresy Pinů za běhu

Koriguje chybu nástroje pro export databází regulátorů EDE Export. Tento nástroj je součástí distribuce systému Desigo XWorks 5.0. Používá se při generování seznamů objektů včetně jejich popisů typů a adres. V případě generování seznamu včetně tzv. Pinů jsou chybně exportovány BACnet adresy pinů. Namísto tečky oddělující pin je exportován apostrof.

Tato volba je implicitně zapnuta pro Desigo 5.0. Volba zkusí po třetím neúspěchu změnit automaticky adresu a zkouší číst znova. Toto velmi zpomaluje komunikaci při kontrole databáze. V případě vypnuté volby a chyby v EDE exportu jsou proměnné s vadnými adresami vyřazeny z obnovy a zalogovány do událostí. Korekce je pak nutno provést manuálně.

Kontrolovat verzi BACnet

Čte a kontroluje z Device Objektu verzi BACnet protokolu (jestli je dobře nastavena verze Desigo 2x, 4.x, 5.x)

Čti vlastnost AckRequired

Potlačení čtení BACnet vlastnosti AckRequired - tato vlastnost u Desigo 4.0 chybí.

Ignoruj BACnet Error rámce

Modul zcela ignoruje příjem chybových rámců protokolu BACnet. Nutné v případě, kdy regulátor generuje nesmyslné chybové rámce (Delta Controls).

Generuj alarm pro resetování

V případě rozšířené poruchy u regulátorů řady Desigo PXC je možné zapnout generování alarmu se stylem Alarm Reset. Tento alarm je generován v případě odeznělé (Status=0) a nekvitované poruchy (Ack=1) u rozšířených poruch. Tento stav obvykle blokuje regulaci a je nutno jej odkvitovat.

Generuj půlnoc v Scheduleru

Modul u regulátorů Desigo 4 přidává půlnoční záznam (kopie předchozího posledního záznamu) tak, ať se Scheduler chová jako dosud a nevypíná o půlnoci regulaci pokud chybí půlnoční záznam.

Manuální režim v Scheduleru

Volba, zda má TSP dialog dovolit manuální ovládání.

Čti výjimky ze Scheduleru

Povolí vyčítání Schedule Exceptions a zpřístupní je přes fieldy Except.XXXX v proměnné typu Schedule. Pro ovládání je nutná speciální ovládací obrazovka.

Ukaž seznam poruch v CommonFault

Při zapnuté volbě ve fieldech CommonFault.Fault.List[10] a .Fault.Used[10] zpřístupní 10 aktivních alarmů. Seznam alarmů obdobně jako v PXM20 s možností filtrů v následných nastaveních.

Texty u alarmů, které modul najde v databázi vezme z databáze (popis proměnné + názvy 2 nadřízených skupin), ostatní vyčte z popisů objektů z regulátoru.

Ukaž ComnAlarms v CommonFault

Implicitně vypnuto, pak nezobrazuje alarmy objektů typů ComnAlarm (protože by byly pravděpodobně hlášeny duplicitně.

84

Ukaž definované alarmy v CommonFault

Implicitně vypnuto, pak nezobrazuje alarmy objektů které jsou definované v databázi a je nastaveno jejich vyhlašování (pomocí fieldu .Status).

Chybou hlavní hodnotu znevalidni

Pokud vlastnost PRESENT_VALUE vrací při čtení BACnet Error znamená to, že hodnota je na podřízené sběrnici dočasně nedostupná a pouze ji považuj za dočasně nevalidní (splity Daikin VRV).

Podpora polí u hlavní hodnoty

V BACnet vlastnosti PRESENT_VALUE podporuje modul hodnoty typu pole (žaluzie u pokojových regulátorů Desigo PXC3).

10.5 Výjimky časového programu v regulátoru PXC

Časový program (Scheduler) umožňuje kromě standardního týdenního zadání určit i Výjimky. Dny výjimek lze definovat jedním ze tří způsobů:

- Datum, resp. Rozsah datumu
- Den, Týden a Měsíc
- odkazem na Globální kalendář

V regulátorech řady PXC jsou standardně využívány tři Globální kalendáře (Cln1, Cln2, Cln3) v Chartu Global. Je-li v segmentu více regulátoru, pak jsou směrodatné Globální kalendáře v regulátoru s příznakem Primární server. Výstup časového programu je dán hierarchií jednotlivých zápisů, především hodnotou Priority, přičemž pro týdenní program je rezervována hodnota 16 a pro prioritu Výjimky lze využít zbytek rozsahu, tzn. <1;15> (implicitně 15).

10.5.1 Implementace výjimek v ProCopu

Standardně Bacnet umožňuje využívat 20 řádků změn v časovém programu (shodně i ve výjimce časového programu) a 20 výjimek. V ProCopu je implementováno 10 řádků a 10 výjimek. Bacnet objekt Schedule v PxM odulu obsahuje extenzi Excp poskytující tato pole:

Název pole	Popis	Významy hodnot	Rozsah pole
Used	existence, resp. způsob definice výjimky	0 = Nepoužito (výjimka neexistuje) 1 = Datum 2 = Rozsah datumu 3 = Den, Týden, Měsíc 4 = odkaz na Globální kalendář Pozn: Zápisem 0 lze smazat Výjimku	<0;9>
Priority	priorita výjimky	pro prioritu Výjimky lze využít rozsah <1;15> (defaultně 15)	<0;9>
GlobCal	číslo Globálního kalendáře, na který je Výjimka odkazována		
DateStart	datum, nebo počáteční datum rozsahu		<0;9>
DateStop	koncové datum rozsahu		<0;9>
DayOfWeek	den v týdnu	0 = Každý, 1 = Pondělí,, 7 = Neděle	<0;9>
WeekOfMon	tý den měsíce	0 = Každý 1= 1. až 7. v měsíci 2 = 8. až 14. v měsíci 3 = 15. až 21. v měsíci 4 = 22. až 28. v měsíci 5 = 29. až 31. v měsíci 6 = Posledních 7 dnů měsíce	<0;9>
Month		0 = Každý $1 = Leden, \dots 12 = Prosinec$	<0;9>
EvnTime	denní čas provedení změny		<0;99>
EvnStep	požadovaná hodnota změny	stupeň, žádaná (dle typu Sheduleru)	<0;99>
EvnUsed	existence EvnTime, EvnStep	Pozn: Zápisem FALSE lze smazat zápis	<0;99>

Globální kalendář má obdobná pole pro definici výjimkových dnů:

Název	Popis	Významy hodnot	Rozsah pole
Used	existence, resp. způsob defiinice vyjímkových dnů	0 = Nepoužito 1 = Datum 2 = Rozsah datumu 3 = Den, Týden, Měsíc Pozn: Zápisem 0 lze smazat Výjimku	<0;9>
DateStart	datum, nebo počáteční datum rozsahu		<0;9>
DateStop	koncové datum rozsahu		<0;9>
DayOfWeek	den v týdnu	0 = Každý, 1 = Pondělí,, 7 = Neděle	<0;9>
WeekOfMon	tý den měsíce	0 = Každý 1= 1. až 7. v měsíci 2 = 8. až 14. v měsíci 3 = 15. až 21. v měsíci 4 = 22. až 28. v měsíci 5 = 29. až 31. v měsíci 6 = Posledních 7 dnů měsíce	<0;9>
Month		0 = Každý 1 = Leden, 12 = Prosinec	<0;9>

🞽 Poznámka

Každoroční výjimky lze definovat rokem 1980, ale jen jako jednoduchý datum, nikoliv jako rozsah.

Při pokusu o zápis každoročního rozsahu ProCop použije akt. rok (ev. rok již zadaný v počátečním datumu, je-li zadáván rok 1980 jen v koncovém datumu).

10.5.2 Aplikace výjimek ve vizualizačním projektu

Ve vlastnostech PxModulu, v sekci "Podrobná nastavení" najdeme volby:

- 1. Čti výjimky ze Scheduleru po zapnutí budou výjimky čteny z regulátoru při komunikaci
- 2. Vytvářej proměnné kalendářů nutno zapnout před použitím funkce Autogenerace (resp. Autodiference), požadujeme-li vygenerovat Globální kalendáře

K vizualizaci výjimek je připravena šablona displeje "PX Výjimky scheduleru", která již obsahuje vložený knihovní atributový blok "Výjimky PXC Scheduler Step". Pro vizualizaci výjimek analogového časového programu lze jednoduše blok zaměnit za "Výjimky PXC Scheduler W". Uvedené knihovní bloky vizualizují tři Globální časové programy (Cln1, Cln2, Cln3) a Výjimky konkrétního časového programu.

🞽 Poznámka

Výše uvedeným způsobem vytvořenému displeji výjimek je třeba zapnout příznak "Dialog".

10.6 SNMP modul

V této části je popsán protokol SNMP a jeho příslušný modul dostupný v Panel nástrojů ProCop od verze 3.7.

10.6.1 SNMP protokol

SNMP

Snmp protokol slouží k monitorování jednotlivých zařízení v síti. Příklady:

Tiskárna

- zbý vající toner
- o offline/online stav
- chybějící papír
- zaseknutý papír
- o restartování tiskárny



• HP server

- o volná kapicita logických disků
- $\circ~$ stav diskových polí
- o stav ventilátorů
- o teplotní stav serveru



- Cisco switch
 - o stav jednotlivých rozhraní
 - o rychlost rozhraní
 - o doba běhu

					mon	itoring	cisco s	witche						
Název: Popis: Doba bě Lokace:	Cise SG3 hu: 692 Ser	coSwitch 350-28P2 2:35h verovna	n3 28-Port (horni P	Gigabit OE swit	PoE Man ch	aged Sv	witch							
Down	Down	Up	Up	Up	Up	Up	Up	Down	Down	Down	Up	Down	Down	
Up	Down	Up	Up	Up	Down	Up	Down	Down	Down	Up	Up	Down	Down	

SNMP verze

Snmp protokol existuje ve třech různých verzích (1, 2, 3).

- Verze 1 Použito v modulu Panelu nástrojů ProCop od verze 3.7. Slouží většinou jen pro čtení proměnných ze zařízení. Není šifrováno ani zabezpečeno autentikací.
- Verze 2 Tato verze obsahuje navíc autentikaci.
- Verze 3 Obsahuje navíc šifrování.

Agent a manager

Monitorovaná strana (síťové zařízení) obsahuje **agenta**, který odpovídá na dotazy monitorovací strany neboli **managera** (SNMP modul v Panelu nástrojů ProCop).

Trap režim

Kromě **standartního režimu**, ve kterém manager odešle dotaz agentovi a ten mu vrátí odpoveď s proměnnou, ještě existuje režim *Trap*. V tomto režimu *agent* rovnou posílá paket na *managera* bez předcházejícího dotazu. Je využíván zejména při náhlý ch událostech např.: vypnutí zařízení, selhání některé funkce atd.

Protokol pro přenos dat a síťový port

Přenos dat zajišťuje prokol UDP. Port 161 je vyhrazen pro standartní režim a 162 pro režim Trap.

OID - Object identifier

Každá proměnná v *SNMP* je jednoznačně identifikována pomocí číselného identifikátoru **OID**. Ten je tvořen posloupnosti čísel jeho nadřazených OID oddělených tečkou.

MIB - Management Information Base

Je to textový soubor který popisuje celý strom proměnných a jejich metadat. **MIB** není nezbytně nutný pro konfiguraci kanálů v Panelu nástrojů ProCop, pokud již známe jednotlivé OID, datové typy i jejich významy. V opačném případě MIB slouží k vyhledání daného OID, jelikož obsahuje i textový popis dané proměnné a další metadata.

Pro procházení MIB databáze je vhodné použít například program Ireasoning MIB browser.



Na obrázku lze vidět proměnná sysUpTime s OID 1.3.6.1.2.1.1.3. Důležité metadata pro proměnné v Panelu nástrojů ProCop jsou:

- Name unikátní název proměnné
- Syntax datový typ proměnné
- OID identifikátor

Tento OID je postupně složeno z těchto předků: 1 ISO 1.3 ORG 1.3.6 Department of Defense 1.3.6.1 Internet 1.3.6.1.2 IETF Management 1.3.6.1.2.1 SNMP MIB-2 1.3.6.1.2.1.1 SNMP MIB-2 1.3.6.1.2.1.1.3 sysUp Time

Datové typy

datový typ SNMP	datový typ v ProCop	minimum	maximum
Integer	Analog	-2147483648	2147483647
Integer32	Analog	-2147483648	2147483647
UInteger32	Analog	0	4294967295
Counter32	Counter	0	4294967295
Counter64	Counter	0	18446744073709551615
TimeTicks	Counter	0	4294967295
IpAddress	Text	-	-
Object identifier	Text	-	-
Octet String	Text	-	-

Typy dotazů

V SNMP je několik typů dotazů, zde jsou uvedeny dotazy použité v SNMP modulu Panelu nástrojů ProCop

- GetRequest Dotaz managera na agenta na jedno nebo více OID.
- GetNextRequest Manager pošle *OID* a agent mu vrátí následující *OID* ve stromové struktuře *MIB*. Pokud pošleme dotaz na *OID* 1.3 a z odpovědi uděláme další dotaz, můžeme takhle projít celou strukturu *MIB* v jednoduchém cyklu. Tento způsob použitý při autokonfiguraci se nazývá "*Walk*".

10.6.2 Nastavení modulu

Vložení a nastavení komunikační cesty HTTP

V Panelu nástrojů ProCop otevřeme v režimu Designer Správu projektu, vybereme Cesty a moduly a stiskneme klávesu Insert, nebo symbol zeleného plus. Vybereme UDP a potvrdíme.

ঢ় Vložit	×
Parametry Typ:	UDPWay 🔻
Info:	Cesta pro komunikaci přes IP/UDP
Verze:	3.00.00

Vložení komunikační cesty UDP

- Ve vlastnostech a dynamizacích cesty nejprve nastavíme název a v části Síť, v položce UDP port nastavíme příslušný port. V našem případě 161 pro standartní režim a 162 pro režim *Trap*.
- Do pole Adresa v části O cestě nastavíme adresu SNMP managera.

Vlastnosti a dynamizace komunikační cesty UDP

	Hodnota			
🖬 Název	UDP cesta pro SNMP			
🖬 Popis				
Scope	<inherited></inherited>	v		
🗄 0 cestě	UDPWay			
💷 Zakázáno	Ne	¥		
Adresa (X.X.X.X)	192.168.0.10			
🖬 Spojení zpět (IP, tel. čísl				
🗆 Síť				
UDP port	161	¥		
🔲 Maximálně spojů	8			
🗆 Skupinové vysílání				
🔲 Povolit skupinové vysí	. 🔲 <ne></ne>			
Adresa skupiny	225.0.0.1			
🖪 Adresa rozhraní	0.0.0.0	¥		

• Na vloženou UDP cestu přidáme modul SNMP.

Vlastnosti a dynamizace modulu SNMP

🤹 Vlastnosti a dynamiza	ace
	Hodnota
🖩 Název	cisco switch 220
🗉 Prefix	CISW220
🗉 Popis	
🗉 Scope	<inherited></inherited>
Soubor	snmpSw220.pio 🔤
🗉 Zakázáno	Ne
🗄 O modulu	SnmpModule
Parametry komunika	
💷 Permanentní obnova	1h
💷 Počet opakování	3
Čekat na odpověď	3 s
🔲 Styl alarmu komunikace	<comm alarm=""></comm>
🗆 Perioda obnovy	
Perioda (0=vyp)	1 m
🔲 Posunutí periody	0 s
🔲 Obnov při startu	✓ <ano></ano>
Validita (0=vždy)	0 s
První obnova (0=o	0 s
Poslední obnova (1	1 d
🗆 Parametry spojení	
Pokusů pro navázání	3
Tel. číslo, adresa …	192.168.1.220
🔲 Komunikační oblast	
Parametry SNMP	
Trap režim	🔲 <ne></ne>
Autokonfigurace	Ano>
🔲 Max dotazů v paketu	1
🔲 Autokonfigurace kanálů	Autokonfigurace
Výsledek	

- Modul pojmenujeme.
- V části Parametry spojení do pole Tel. číslo, adresa vložíme IP adresu agenta.

Trap režim

V případě trap režimu adresu není třeba vyplňovat. Pouze zaškrnout checkbox Trap režim. Tento modul bude přijímat *Trap* pakety ze všech zařízení a IP adres. V projektu je nutno mít pouze jeden tento modul. V případě příjmu trap paketu, je tento paket předán dle ip adresy příslušnému modulu ve standartním režimu, který jej dále zpracuje.

Autokonfigurace

Autokonfigurace slouží pro vyčtení všech proměnných ze zařízení.

- Postup:
- Zaškrtneme checkbox Autokonfigurace.
- Spustíme modul a v řízení modulů a komunikačních cest uvidíme hlášku "Načítám seznam proměnných", po dokončení vyčtení se zobrazí "pauza hh:mm:ss".
- Zastavíme modul.
- Klikneme na tlačítko Autokonfigurace.
- Po úspěšném dokončení autokonfigurace se zobrazí v poli Výsledek: "Ok, xx proměnných", kde "xx" značí počet vyčtený ch proměnných.
- V případě neúspěšné autokonfigurace bude zobrazena hláška s příslušnou chybou.

Max dotazů odeslaných najednou

Pokud chceme ušetřit počet odesílaných paketů, je možnost nastavit víc dotazů v jednom paketu.

10.6.3 Nastavení proměnných

V této kapitole je popsáno nastavení agenta, nalezení jednotlivých proměnných a jejich následné vložení do monitorovacího programu ProCop.

🗾 Poznámka

Tento postup je popsán pro Cisco Switch. Pro jiná zařízení se bude lišit, ale je možno jej použít jako obecný postup.

Nastavení agenta v zařízení

V zařízení, které chceme monitorovat je nejprve třeba nakonfigurovat SNMP agenta. **Příklad pro nastavení agenta v Cisco S witchi:**

- Ve webovém rozhraní switche v SNMP/ Communities přidáme do Community Table nový záznam.
- Do IP adresy vložíme adresu managera (SNMP modul panelu nástrojů ProCop).
- Do Community string napíšeme "*public*". *Community string* může být libovolný řetězec znaků sloužící k jednoduché, nešifrované autentikaci. Výchozí string pro SNMP modul je "*public*".

Webové rozhraní pro nastavení agenta



• V SNMP/ Notification Recipients SNMPv1,2 přidáme nového příjemce *Trap paketů* s adresou managera a portem 162. Číslo portu se zde musí shodovat s portem v SNMP modulu Panelu nástrojů ProCop.

Nastavení Trap paketů

92

P A	dd SNMP Notification Recipie	ent - Google Chrome
i	Nezabezpečeno 192.1	68.1.218/cs20f36cde/snmpNote/snmp_notification_receiver_a.htm
	Server Definition:	By IP address By name
	IP Version:	Version 6 Version 4
	IPv6 Address Type:	Iink Local Global
	Link Local Interface:	VLAN 1 x
•	Recipient IP Address/Name:	192.168.1.17
•	UDP Port:	162 (Range: 1 - 65535, Default- 62)
	Notification Type:	Traps Informs
8	Timeout	15 sec (Range: 1 - 300 Default: 15)
	- · · ·	15 See (Range, 1- See, Belault, 15)
*	Retries:	3 (Range: 1 - 255, Default: 3)
	Community String:	public 🔻
	Notification Version:	SNMPy1
	Notification version.	SNMPv2
	Notification Filter:	Enable
	Filter Name:	Ŧ
	Apply Close	

- Nyní by měl být SNMP agent v Cisco Switchi nakonfigurován.
- Správnost konfigurace SNMP agenta můžeme nyní ověřit autokonfigurací v SNMP modulu Panelu nástrojů ProCop.

🕭 Důležité upozornění!

Webové rozhraní k nastavení SNMP je rozdílné v každém typu zařízení. Pokud tento návod nezafungoval na cisco switch nebo konfigurujeme jiné zařízení, zkusíme se řídit oficiálním manuálem.

Postup nalezení OID pro standartní režim

Uvedený příklad pro Cisco Switch:

Nejdříve si určíme co chceme monitorovat nebo zobrazit.

- název
- popis
- lokaci při větším množství switchů nám pomůže ho fyzicky lokalizovat

🞽 Poznámka

Tyto 3 výše zmínění proměnné by měl nastavit administrátor routeru při jeho konfiguraci. My si je jen zobrazíme pro lepší orientaci při velkému počtu podobných zařízení.

- dobu běhu
- nastavené stavy rozhraní (up/down)
- aktuální stavy rozhraní (up/down)
- rychlosti rozhraní

Postup nalezení OID:

94 Tipy a triky

- Nainstalujeme program pro procházení MIB souborů např.:Ireasoning MIB browser.
- Stáhneme z internetu MIB soubor RFC1213-MIB.mib. V tomto souboru jsou definované naše hledané OID spolu s metadaty.
- V programu načteme MIB soubor, abychom mohli poté lépe identifikovat proměnné.

🚖 Tip!

Pro stažení různých MIB souborů je vhodná například tato stránka.

🞽 Poznámka

Pokud hledáme OID pro jiné zařízení, potřebujeme použít jiný MIB soubor. Ten by měl být na webu výrobce, případně ho zkusíme kontaktovat nebo vyhledat na internetu.

- Do pole adresy zadáme adresu switche.
- Do pole OID zadáme 1.3.
- Jako volbu v operations zvolíme Walk.
- Klikneme na tlačítko GO.

MIB browser

ø	Reasoning MIB Browse	r		_	-	- 0 ×
Fi	e Edit Operations	Tools Bookmarks H	elp			
Add	ress: 192.168.1.218	✓ Advanced C	D: 1.3	• Operations:	Walk 👻	G 0
SN	IP MIBs		Result Table			

Pokud jsme vše nastavili správně, měla by se objevit tabulka všech proměnných v daném switchi. Jestliže se v načetlém MIB souboru vyskytují nějaké proměnné z vyčtené walk tabulky, tak místo OID je zde zobrazeno jméno. Dané jméno by nám mohlo pomoct v určení významu dané proměnné (př.: *sysUptime* by mohl znamenat "systémový čas běhu"). Vlevo dole vidíme její další metadata.

Dle názvu některých z prvních proměnných (označených žlutě na obr.) usoudíme, že se jedná o naše hledané proměnné k popisu switche.

Při pohledu na další proměnné v tabulce narazíme na periodicky se opakující počet proměnných s prefixem "*if*", který značí interface (rozhraní) a s názvy *Index, Descr, Speed, AdminStatus, OperationStatus*. Tyto proměnné potřebujeme k monitoringu jednotlivých rozhraní.

🚳 iReasor	ning MIB Browser				_ Θ Σ
File Edit	t Operations Tools Bookmarks Help			-	
Address: 19		1.3.6.1.2.1.1.6.0	 Operations 	: Walk	🔹 🌈 Go
SNMP MIBs		Result Table			
🌳 MIB Tre	e 🔺	Name/OID	Value	Type	TP:Port ⊽
🖻 🕛 iso.(org.dod.internet	sysDescr 0	0x53 47 33 35 30 2D 32 38 50 20 32 38 2D	OctetString	192 168 1
- <u>e</u>	directory	sysObjectID.0	1.3.6.1.4.1.9.6.1.95.28.5	OID	192, 168, 1
🖻 👘 👘	mgmt	syst InTime 0	695 hours 12 minutes 33 seconds	TimeTicks	192 168 1
<u> </u>	📕 mib-2 📃	sysContact.0	oso nodio 12 mindres os secondo	OctetString	192, 168, 1
	🖃 🎍 system	sysName.0	CiscoSwitch3	OctetString	192, 168, 1
	sysDescr	sysLocation.0	0x53 65 72 76 65 72 6F 76 6E 61 20 68 6F	OctetString	192, 168, 1,
	sysObjectID	sysServices.0	6	Integer	192, 168, 1
	sysUpTime	.1.3.6.1.2.1.1.8.0	0 millisecond (0)	TimeTicks	192.168.1
	sysContact	.1.3.6.1.2.1.1.9.1.2.1	.1.3.6.1.4.1.89.73	OID	192, 168, 1
	sysName	.1.3.6.1.2.1.1.9.1.3.1	RS capabilities	OctetString	192.168.1
	sysLocation	.1.3.6.1.2.1.1.9.1.4.1	0 millisecond (0)	TimeTicks	192, 168, 1
	sysServices	ifNumber.0	40	Integer	192.168.1
	Interfaces	ifIndex.1	1	Integer	192, 168, 1
	ifNumber	ifIndex.2	2	Integer	192, 168, 1
	If Table	ifIndex.3	3	Integer	192.168.1
	E mitentry	ifIndex.4	4	Integer	192, 168, 1
	ifindex	ifIndex.5	5	Integer	192.168.1
	inDescr	ifIndex.6	6	Integer	192.168.1
	ift ype	ifIndex.7	7	Integer	192.168.1
		ifIndex.8	8	Integer	192.168.1
Name	sysLocation	ifIndex.9	9	Integer	192.168.1
	.1.3.6.1.2.1.1.6	ifIndex.10	10	Integer	192.168.1
MIB	RFC1213-MIB	ifIndex.11	11	Integer	192.168.1
SyntaX	Displaysting (OCTET STRING) (SIZE (0255))	ifIndex. 12	12	Integer	192.168.1
Access	reau-write	ifIndex.13	13	Integer	192.168.1
Status Defi/al	manuatory	ifIndex. 14	14	Integer	192.168.1
Dervai		ifIndex. 15	15	Integer	192.168.1
Indexes		ifIndex. 16	16	Integer	192.168.1
	The physical location of this node (e.g.,	ifIndex. 17	17	Integer	192.168.1
Descr	telephone closet, and hoor).	ifIndex. 18	18	Integer	192.168.1
		ifIndex. 19	19	Integer	192.168.1
		ifIndex.20	20	Integer	192.168.1

MIB browser a walk tabulka

Zdá se, že jsme nalezli všechny potřebné proměnné i jejich OID a můžeme je vložit do proměnných v našem SNMP modulu.

- První možností je použití autokonfigurace ⁸⁹ a poté vymazat nepotřebné kanály. Tato možnost nebude ideální, protože celkový počet proměnných v jednom zařízení múže být v tísících.
- Druhou možností je ruční vložení kanálů.

Obě možnosti jsou popsány v předchozí kapitole 89.

Postup nalezení OID pro režim Trap

Uvedený příklad pro tiskárnu značky OKI C531.

- Stáhneme si MIB určený pro naši tiskárnu OKIDATA-MIB.mib.
- V MIB browseru načteme příslušný MIB.
- Ve výpisu stromové struktury jsou na konci uvedeny Trap proměnné.
- Všimneme si, že všechny proměnné mají stejné OID, ale liší se v parametru *Specific*. K našim proměnným v modulu SNMP nám stačí tento parametr a OID. Tento trap nenese žádnou hodnotu, pouze značí aktuální stav zařízení pomocí parametru *specific*.

\delta iRea	isoni	ng MIB	Brows	er	<u> </u>			_	_
File	Edit	Oper	ations	То	ols	Bookm	arks	Help	-
Address:				•		Advanced	I	OID:	.1.3.6
SNMP M	IBs								
MIB	Tree so.or	g.dod.i irectory gmt xperime rivate NMPv1 trapF trapF trapF trapF trapF	nternet ental TRAPs PrinterOf PrinterOf PrinterPa PrinterPa PrinterPa PrinterPa	nline ffline rAtta aper(aper) oorO ror	ache ow Dut Jam pen	d			
Enterpris	se	printSe	rvers (.	1.3.6	w 5.1.4	4.1.2001.	1.2.68	3.6)	
Generic		6			-				
Specific		4							
Variables	S	output	Index						
Descr		"The pr	rinter to	ner is	s lov	ı."			

Zobrazení Trap proměnných v MIB browseru

- Nalezené Trap proměnné vytvoříme v modulu SNMP Panelu nástrojů ProCop. Jelikož mají všechny stejné OID a liší se v paramteru "Specific", tak budou zastoupeny jednou proměnnou a její tabulkou.
- Ve vlastnostech a dynamizacích proměnné nastavíme příslušné OID a klikneme na "*Specific-trap tabulka"*. Do druhého sloupce vypíšeme všechny parametr "Specific" ze všech Trap proměnných, které chceme monitorovat. Jako formát výpisu nastavíme "*<User Table>*", kde první sloupce s názvem "hodnota" odpovídá prvnímu sloupci z

Tipy a triky

"Specific-trap tabulky". Druhý sloupec je textový popis Trapu.

対 Tip!

Trap proměnná přímo vybízí nastavení alarmu, aby se nám všechny náhlé události ihned zobrazovaly ve výpisu alarmů.

10.7 Mailer modul

Pro odesílání e-mailu Monitorovacím systémem ProCop je k dispozici licencovaný modul Mailer. Lze vložit na komunikační cestu TCP Way, kde postačí nastavit příslušný TCP port mailového SMTP serveru, obvykle 25. Pokud je komunikace zabezpečena, je nutné nastavit ve vlastnostech cesty typ zabezpečení, nebo režim automatický.

K čemu lze modul Mailer použít:

- odeslání alarmu e-mailem, aktivní skupině příjemců, nebo jednotlivci
 - o lze definovat pomocí filtrů dynamizace OnAlarm 68, které alarmy budou posílány
 - aby nedocházelo k příliš častému zasílání vzniků a zániků alarmů, lze použít metodu zpoždění odesílání alarmů 70
- odeslání libovolného textu, například s technologickými hodnotami e-mailem, aktivní skupině příjemců, nebo jednotlivci

🗾 Poznámka:

Modul slouží pouze pro odesílání e-mailů, jedná se čistě os SMTP komunikaci, žádné e-maily nelze tímto modulem přijímat.

Parametry modulu Mailer

Perioda obnovy

Nastavení určuje periodu vyzkoušení, zda SMTP server reaguje a lze se k němu účtem připojit. Slouží pro kontrolu připravenosti k odeslání mailu. Pokud se nepodaří některý e-mail odeslat, s touto periodou modul zkouší tento e-mail odeslat znovu.

🞽 Poznámka

Tuto periodu nenastavujte na příliš krátkou, může se stát, že dle nastavení SMTP serveru bude po nějakém čase tento zdroj e-mailu odmítán, jako ochrana před spamováním, nebo přetížením.

Parametry spojení, Paremetr (SMTP Server)

Jedná se o IP adresu, nebo DNS jméno poštovního serveru, který má sloužit k odesílání pošty.

Prefix odesílaných zpráv

Text, který je připojen na začátek těla každého e-mailu. Vhodné zejména pro odesílání alarmů a více zdrojů.

Logování odesílaných zpráv

Volba a případně styl záznamu všech odesílaných zpráv. Obvykle se odeslání každého e-mailu zapisuje do systémového zápisníku.

Parametry e-mailu

- Doména odesílatele mohou požadovat SMTP servery hned při oslovení (nemusí být vyplněna)
- Adresa odesílatele e-mail, ze kterého má být zpráva odeslána
- Adresa odpovědi e-mail, na který má uživatel odpovědět (nemusí být vyplněn)
- Předmět e-mail Textová hlavička e-mailu
- Max počet příjemců (0=neomezeně) nastavitelný limit maximálního počet příjemců
- Kódování Mailer modul umožňuje zprávu překódovat do požadované znakové sady
- Diakritika implicitně se odstraňuje ze zprávy diakritika, tuto volbu lze vypnout dle potřeby

🍕 Vlastnosti a dynamiza	ace 🛛		
	Hodnota		
🖩 Název	MailerModul		
🗉 Prefix	MAIL		
🗉 Popis	Emailový modul		
🗉 Scope	<inherited></inherited>		
Soubor	MailerModul.pio		
🗉 Zakázáno	Ne		
🗄 O modulu	Mailer Module		
🗆 Perioda obnovy			
🖬 Perioda (0=vyp)	5 m		
💷 První obnova (0=od p	0 s		
🖪 Poslední obnova (1d=	1 d		
🗆 Parametry spojení 👘			
Parametr (SMTP Server)	192.168.1.17		
🗆 Parametry pageru			
💷 Prefix odesílaných zpráv	Ukázkový projekt:		
🗄 Logování odesílaný…			
🗆 Parametry e-mailu			
💷 Doména odesilatele	domena.cz		
💷 Adresa odesilatele	mail@domena.cz		
Adresa odpovědi			
💷 Předmět e-mailu	ProCop e-mail		
🔲 Max počet příjemců (0	0		
🔲 Kódování textu	Latin2 (ISO 8859-2)		
🔲 Diakritika	Ano>		

Vlastnosti a dynamizaci proměnné modulu Mailer

🗾 Poznámka:

V současné verzi Mailer modul neumožňuje komunikovat s SMTP serverem, který vyžaduje autentizaci, nelze tedy zadat uživatelské jméno a heslo pro připojení k serveru.

Systémové proměnné

Modul mailer má mimo jiné následující dvě systémové proměnné:

- ActiveGroup definuje aktivní skupinu odesílatelů
 - Message zasílaný text zprávy

Popis proměnných a chování

V sekci Proměnné Mailer modulu vložíme proměnné každou pro jednoho příjemce e-mailů. Parametry zprávy

- Adresát e-mailová adresa příjemce
- Popis jméno, nebo popis skupiny příjemce pro vlastní potřebu a identifikaci
- Skupina pokud využíváme přepínání cílových skupin příjemců, pak se definuje členství ve skupině

Hromadné zasílání jako

Definuje, do které části hlavičky e-mailu bude adresát vložen.

🖏 Vlastnosti a dynamizace 📮				
	Hodnota			
💷 Název	MAIL.D1			
Popis	Dispečer 1			
💷 Použito	Ne>			
🗆 Parametry zprávy				
🔲 Adresát	dispecer1@domena.cz			
Popis	František Dispečer			
🔲 Skupina	1			
🔲 Hromadné zasílání jako	To: (Komu)		J .	
🗆 Limity, přepočty				
🗉 Dolní mez	-1E06			
🔲 Horní mez	1E06			
Násobitel	1			
🗉 Posun	0			
🔲 Jednotka		v		
Formát výpisu	<default></default>	v		
Styl záznamu	Set Value	•		
🗄 Alarm				
🗄 Trend				

Dynamizace emailového modulu

Odesílání e-mailu

Nejjednodušší způsob odeslání e-mailu je, že vytvoříme jednu proměnnou, nastavíme ji parametry zprávy a dynamizací do ní přiřadíme text e-mailu. který má být odeslán.

Mailer modul umožňuje odesílat e-maily předvolené skupině uživatelů. Každá vložená proměnná reprezentuje jednoho adresáta, který je členem číselně označené skupiny.

Pokud se do systémové proměnné Message přiřadí text správy, je odeslána všem uživatelům, kteří mají nastavenu číselnou skupinu na stejnou hodnotu, jako systémová proměnná ActiveGroup. Pro nastavení členství ve skupinách, jmen a e-mailu adresátu a přepínání aktivní skupiny je vhodné vytvořit speciální displej.

Odeslání alarmů e-mailem lze provést jednodu3e dynamizaci OnAlarm 68 a přiřazením do systémové proměnné @.M essage výsledku funkce CmdLineAlarm()

💢 Tip!

Projekt s příkladem modulu Mailer, dynamizace OnAlarm a skriptu na odeslání technologické hodnoty e-mailem naleznete v sekci ke stažení, ProCop 3.7 pod názvem Tips and Tricks Sample 1.

99

Index

-~-

~` 21

- A -

Adresa serveru 27 Aktivní 29 Aktualizace hodnot 27 Alarmní styly 67 ANSI 8 Automatická 28

- B -

Baud Rate 12 bigint 31 bit 31 Buňka 25

- C -

Consumption 30 ConsumptionValid 30

- D -

databáze 25 Date 30 DateOrNow 30 datetime 30.31 DateTimeOrNow 30 Datové typy 31 DE-211 7 DE-311 7 Definice exportních tabulek 28 Definice struktury tabulky 29 Definice tabulek 28 délka rámce 12 Dispečerské PC 8

- E -

EEPROM 13 Ethernet / RS232 7 Ethernet to serial 11 Exit 13

- F -

Firewall 15 float 31

© 2022 ALFA Mikrosystémy, s.r.o.

Force transmit 11

- G -

Gateway 9

- H -

heslo 9 Hlavní nabídka 9 Hyperterminál 8 char(64) 31

- | -

Index 25 Instance 25 int 31 IP adresa 8

- K -

komunikační chyby 15 komunikační rychlost 12 Konfigurace IP adresy NPort 9 Konzola správy ProSQL 27

- L -

Local listen Port 11 Logické adresy 13

- M -

maska sítě 8 MBus 11

- N -

Nastavení sériového portu 12 Nastavení UDP Server/Client režimu 8 nastavení Windows Firewall 15 nastavitelný čas čekání 7 nastavitelný počet pokusů 7 Název sloupce 29 Názvy sloupců v definici řádků tabulky 31 Netmask 9 none 31 Normal 28 Note 30 NPort 7 **NPortWay** 13

- 0 -

obnovení databáze 48

100

Odkazy mezi sloupci 31 Odkazy mezi sloupci při definici řádků 31 odstranění databáze 48 OnCommand 61 OP_mode 11 OverallValid 30 OVádací panely 15

- P -

Parametrech spojení 13 Parametry modulu PX 78,86 Parametry připojení k datovému serveru 27 Parametry připojení k SQL databázi 27 Parametry spuštění ProSQL 27 parita 12 Password 9 PComm Lite Terminal emulator 8 ping 15 Plánování úlohy 28 Počet opakování 13 počet stop bitů 12 Popis exportu měřičů spotřeb 33 Požadavky před realizací 25 Prefix 30 Primární index 25.29 27 Primární klíče Problémy s virtuálními porty 7 Projekt ProSQL 27 ProSQL 25 Předvolba 29 Předvolené hodnoty v definici řádků 31 Přepínače na převodníku 8 Příklad exportu měřičů spotřeb 33 PX vlastnosti 78.86

- R -

31 real reindex 54 54 reindexace Report 28 Report Tabulka 31 restore 48 restore databáze 48 rozsáhlé sítě 7 RS422 7 RS485 7 rychlost komunikace 13 Řádek 25 Řešení problémů s komunikací 15 Řetězec připojení 27 řízení toku 12

- S -

Send Command 61 Serial to Ethernet 11 Serialport 12 serverConfig 9 SetValue 61 54 shrink Síť 13 Sloupec 25, 29 Složený index 25 smallint 31 spolehlivější komunikace 7 Spuštění a kontrola výsledku 33 SQL 25 sql nástroje 55 SQL server 25 SQL server (HW) 25 SQL server (SW) 25 sql tools 55 29 SQL Typ Static IP 9 Struktura dat 25 Struktura databáze 25 Struktura tabulky 25 SW1 8 SW2 8 SW3 8

- T -

25 Tabulka TaskTime 30 telnet 8 31 text Time 30 tinyint 31 Trend 30 TrendBeginTime 30 TrendEndTime 30 TrendValid 30 Typ hodnoty 29 Typy hodnot 30

- U -

UDP port 8, 11, 13 UDP Server/Client 11 UDP Server/Client mode 7 Uložení konfigurace do převodníku 13 Unknown 30 UpdateTime 30 Úpravy monitorovacího projektu 13

		Index	101
UserData	30		7

- V -

Valid 30 Value 30 7 Virtuální port Vkládání, kopírování a mazání definic řádků 31 Vlastnosti tabulky 28 Vlastnosti tabulky - plánování úlohy 28 Volba a konfigurace UDP režimu 11 Volba terminálu po připojení 8 VT100 8 výchozí brána 8 výjimky na firewallu 15 vyrovnávací paměti 12 Vytvoření tabulky 28 Vytvoření tabulky - úlohy 28 Vzdálené připojení přes síť 8

- W -

Windows Firewall 15 WriteTime 30

- Z -

Zadání prázdné proměnné 31 Zařízení nekomunikují 15 změnšení databáze 54 zotavení komunikace 7 Zpoždění alarmů 67