

PRODUKTY TRENDY TECHNOLOGIE



internetové technologie v průmyslové automatizaci

automatizace budov ... EIB ... USB ... HTTP

sběr dat, řízení a vizualizace průmyslových procesů



Nová generace programového systému **Control Web 5** je již na trhu déle než rok. Dnes již můžeme říci, že byla tvůrci programového vybavení pro průmyslovou automatizaci přijata pozitivně. Velmi nás to těší - a intenzivně si uvědomujeme náš závazek dodávat stoprocentně spolehlivý a kvalitní software. Součástí druhé edice produktu je již vestavěný Service Pack 4 (právě aktuální opravný balíček je vždy možno stáhnout zdarma z našeho www serveru), který přináší korekce a opravy podle zkušeností získaných s více než ročním používáním systému v náročných trvalých provezech.

Verze 5 přináší takovou spoustu nových technologií a nových možností, že dodneška je ve většině aplikací z nich využívána jen relativně malá část. Aplikace jsou často konstruovány tak, jak to bylo běžné pro předchozí generaci systému. Nové možnosti současné verze mohou nejen velmi zvýšit kvalitu a vzhled aplikace, ale mohou také ušetřit hodně práce při její tvorbě. Co se dříve muselo pracně programovat, jde dnes téměř samo.

Všechno asi chce svůj čas. Zdá se, že **Control Web 5** má velkou většinu morální životnosti teprve před sebou. A to je koneckonců celkem dobrá zpráva.

V tomto vydání našeho občasníku si kromě jiného ukážeme několik nasazení systému, která jsou zajímavá svými internetovými technologiemi, datovou komunikací, databázovým modelem, počítačovou grafikou nebo svou celkovou koncepcí a která by mohla být inspirací pro naše další směřování.

Stejně jako v minulých číslech i nyní budeme usilovat o to, aby tento materiál byl pro odborníky v našem oboru zajímavý a přínosný. Aby to nebyla jen snůška prázdných marketingových hesel doplněná prefabrikovanými obrázky z fotobanky.

Bez internetových technologií to již dnes nejde

Internetové technologie - obor, který se jimi zabývá, je často označován jako webové inženýrství - se pozvolna a nezadržitelně stávají trvalými průvodci řadou našich každodenních činností. I v průmyslové automatizaci je stále častěji požadován dohled a servis na dálku. A právě vytvoření webového rozhraní k již existujícímu aplikačnímu programu je v systému **Control Web 5** dovedeno k maximální jednoduchosti. Popis průvodce pro tvorbu webové aplikace spolu s ukázkou takto vytvořeného rozhraní si můžete přečíst na straně 3.

Datové sekce jsou silným nástrojem pro distribuovaná data

Při vývoji rozsáhlých distribuovaných aplikací můžete ušetřit spoustu práce a času. Chcete vědět jak? Pak si přečtete o datových sekcích na straně 9.

Automatizace budov se systémem DataLab

Koncepce, kdy každá řídicí jednotka je zároveň embedded počítačem připojeným na rychlou lokální síť přináší do automatizace budov, na rozdíl od klasického řešení s řadou samostatných regulátorů a PLC, novou kvalitu. Ukázkou vzorového řešení na tomto principu naleznete na straně 2.

Rozhraní ActiveX pro jednotky systému DataLab

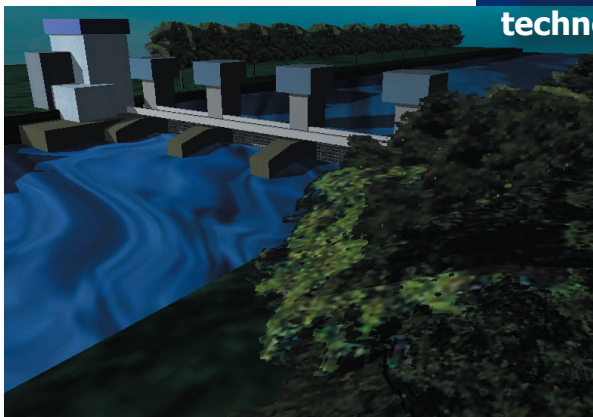
activex Víte, jak efektivně využít programové vybavení, které dostáváte zdarma spolu s každou jednotkou **DataLab**? Komunikaci s jednotkami pomocí dodávaných ActiveX komponent se věnuje článek na str. 6.

Sběrnice EIB v plně automatizované domácnosti

EIB Nízká cena průmyslových počítačů **DataLab** spolu s runtime systémem **Control Web** umožňuje ekonomické používání těchto moderních prostředků při automatizaci rodinných domů a bytů. Podrobnosti naleznete na straně 11.

Nová kvalita vizualizace technologických procesů

Obor počítačové grafiky se v posledních letech proměnil k nepoznání - jsou tu programovatelné grafické procesory svým výpočetním výkonem a datovým tokem převyšujícím CPU.



Paralelní superpočítače ukryté na současných grafických kartách je škoda nechávat nečinně spát.

Ukázky aplikací, které virtuální realitu využívají ke zvýšení přehlednosti a působivosti jsou uvedeny na straně 14 a 15.

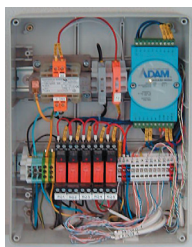


To, že nutnou součástí každé budovy jsou např. rozvody elektřiny a vody, už dávno považujeme za zcela samozřejmé. Nyní se podobně samozřejmými stávají komunikační sběrnice, datové sítě a automatické systémy řízení. Domy vybavené těmito systémy jsou občas označovány jako „inteligentní budovy“. Automatizace řízení vnitřního klimatu a hospodaření s energiemi také již přestává být doménou velkých kancelářských budov a stává se přístupnou i pro rodinné domy i jednotlivé byty. O tom ale až později. Dnešní systémy postavené na průmyslové počítačové technice dokáží mnohonásobně více, než je pouhá regulace a optimalizace spotřeby. Nyní si blíže představme jednu realizaci - kancelářskou budovu státní správy, na které lze ukázat řadu principů, které jsou všem budovám společné.

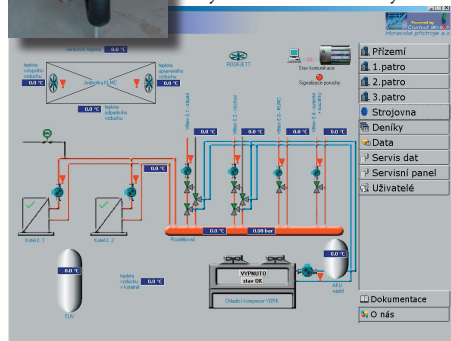
Koncepce řešení

Základ systému měření a regulace tvoří průmyslové počítače **DataLab PC**, umístěné na každém patře budovy a vybavené operačním systémem Windows XP embedded spolu s programovým prostředím **Control Web 5**. Tyto počítače nemají pevné disky, veškeré programové vybavení je uloženo na paměťových Compact Flash kartách. Zpracování vstupně/výstupních signálů je zajištěno externími moduly řady ADAM 4000, které jsou připojeny k jednotkám DataLab prostřednictvím sběrnice RS485. Zde s výhodou využíváme kombinace předností výkonného páteřového Ethernetu spolu s dalekým dosahem a nízkou cenou sběrnice RS485.

Řídicí systém je rozdělen do čtyř rozvaděčových skříní umístěných v jednotlivých patrech objektu. V jednotlivých místnostech v bezprostřední blízkosti ovládaných zařízení jsou umístěny podružné skříně se vstupně/výstupními moduly ADAM.



Řídicí systém ovládá jak klimatizační jednotky umístěné v jednotlivých kancelářích a ventilátory pro odvětrání sociálních prostor, tak centrální vzduchotechnickou jednotku, centrální ventilátor sociálních zařízení a chladicí jednotku, které jsou na střeše objektu.



Řídicí systém také zajišťuje regulaci kotlů a ovládá servoventily okruhů ústředního vytápění.

Řízení klimatizačních jednotek v kancelářských prostorách

Klimatizace v jednotlivých kancelářích zajišťuje teplotní pohodu vytápěním v zimním období a chlazením v letním období. Každá z regulovaných prostor je vybavena teplotním snímačem vzduchu a příložným teplotním snímačem na vratném okruhu jednotky.

Regulační uzel je ovládán přírůstkově podle teplotního spádu na jednotce a rozdílu mezi požadovanou a aktuální teplotou. Požadovaná teplota je nastavitelná samostatně pro každou místnost v týdenním časovém programu na útlumovou a běžnou pracovní hodnotu.



Řízení ústředního topení

Příprava topné vody je zajištěna dvěma kotly provozovanými v kaskádě s ekvitermní regulací teploty topné vody na rozdělovači. Spolu s kotly jsou spínána kotlová čerpadla pro primární oběh topné vody.

Topná soustava zahrnuje čtyři regulované topné větve. Regulované topné větve jsou regulovány pomocí servoventilů na teplotní spád s možností nastavení dvou úrovní teplotního spádu pro plyný a útlumový režim v týdenním programu.

V letním období jsou dvě větve pomocí ovládaných armatur automaticky přepojeny na systém chlazení s centrální chladicí jednotkou.

Systém monitoruje a zpracovává tyto údaje:

- teplota a tlak rozvodu topné vody
- teplota teplé užitkové vody
- teplota vzduchu v kotelně

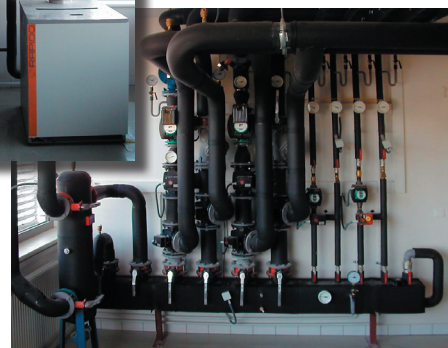


- únik plynu
- teplota a tlak rozdělovače topné vody
- porucha plynových kotlů
- porucha čerpadel a servopohonů

Při vzniku poruchy jsou odstaveny kotle a stav technologie je signalizován obsluze.

Řízení centrální klimatizační jednotky

Centrální nástřešní jednotka zajišťuje přívod teplého vzduchu v zimním období do společných prostor chodeb a schodišť. V letním pak zajišťuje výměnu vzduchu v těchto prostorách. V zimním období je regulována na teplotu na výstupu



z jednotky ve dvou úrovních pro plyný a útlumový provoz v týdenním programu. Systém na jednotce ovládá klapky na přívodu a odtahu s havarijní funkcí, přívodní a odtahový ventilátor a proporcionálně rekuperační uzel. Regulační uzel je přímo jednou topnou větví regulovanou v rámci kotelny.

Systém hlídá následující provozní a poruchové stavy:

- funkce ventilátorů
- protimrazová ochrana na ohříváku
- průchodnost vzduchových filtrů

Řízení ventilátorů sociálních zařízení

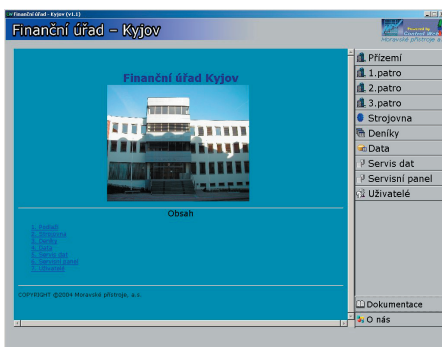
Ventilátory sociálních prostor umístěné v jednotlivých patrech jsou spínány

podle signálů od detektorů pohybu nebo od dveřních magnetický spínačů s nastavitelným doběhem. Centrální nástřešní ventilátor je volitelně spínán od vstupu do větraných prostor s nastavitelným doběhem v kombinaci s možností chodu podle časového programu.

Operátorské pracoviště

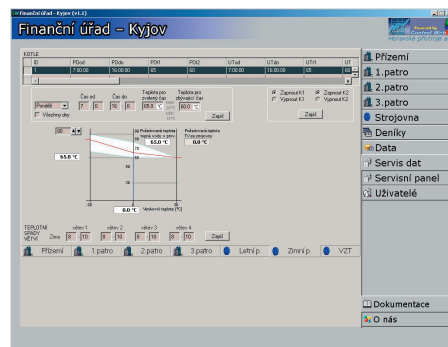
Operátorský terminál je tvořen průmyslovým PC umístěným do rozvaděčové skříně. Operátorské pracoviště sbírá a ukládá data z jednotlivých řídicích jednotek a zajišťuje parametrizaci veškerých řídicích algoritmů. Programové vybavení je stejně jako u všech jednotek **DataLab PC** provozováno v prostředí **Control Web 5** a zajišťuje:

- sběr a vizualizaci technologických dat popisujících aktuální stav strojního vybavení
- archivaci vybraných dat a událostí s možností zpracování formou tabulek a grafů
- zprávu poruchových hlášení
- parametrizaci řídicích algoritmů
- zabezpečení přístupu, autorizaci chráněnou přístupovým heslem
- archivaci vybraných veličin a událostí včetně zásahů obsluhy
- nastavení teplotní pohody i z webových klientů



Vizualizace umožňuje sledovat veškerá data na půdorysech jednotlivých pater objektu. Z operátorského pracoviště je možno zadávat topné křivky, definovat požadované teploty pro řízení klimatizací, nastavovat týdenní programy plného a útlumového režimu pro každou topnou větev i každou klimatizační jednotku jednotku, zadávat teploty v klimatizovaných prostorech a v případě potřeby i blokovat chod jednotlivých technologických celků.

Koncepce, kdy každá řídicí jednotka je zároveň embedded počítačem připojeným na rychlou lokální síť, přináší do automatizace budov, na rozdíl od klasického řešení s řadou samostatných regulátorů a PLC, novou kvalitu. Při srovnatelné ceně, elektrickém příkonu a spolehlivosti ušetříme náklady na zákazkový vývoj a získáváme bohaté možnosti komunikací, konektivitu a vazby na další informační systémy. Systém



může mít internetové rozhraní, konektivitu na databáze, komunikovat přes SMS a GPRS atd.

Nesmíme také zapomenout na roli programového vybavení! Pro kvalitní funkci dnešních automatizovaných budov je nutná dokonalá kooperace všech dříve samostatných technologických celků. Proto nejdůležitější roli začíná v takovýchto systémech hrát programové vybavení. A právě programové vybavení se stává klíčem pro realizaci opravdových inteligentních budov. Použití jediného programového prostředí, systému **Control Web**, pro vývoj i provozování veškerého programového vybavení od řídicích jednotek v rozvaděčích až po operátorské pracoviště s vizualizací přináší uživateli konzistentní, skvěle udržovatelné a snadno rozšiřitelné prostředí a dodavateli podstatné snížení nákladů na vývoj.

Použití webového serveru v průmyslové automatizaci

Pronikání internetových technologií do průmyslových a řídicích aplikací je již nezadržitelné. Zabudované WWW servery se postupně objevují ve stále větším počtu dříve samostatných zařízení.

Výhody používání webového serveru pro vizualizaci technologických dat a pro operátorský přístup k prostředkům průmyslové automatizace jsou zřejmé. Jednak odpadá nutnost na klientské počítače instalovat další software a data jsou tak velice jednoduše přístupná prakticky neomezenému počtu klientů. Další výhodou je platformová nezávislost, která však není v prostředí České republiky tak podstatná (velká většina uživatelů používá „stejný“ operační systém).

Pokud webový server běží přímo na „jednoduchém“ hardware (jako je například průmyslový automat), obvykle poskytuje data pouze v základní pevně definované podobě. Také nebývá možná jejich další archivace ani složitější zpracování. Kromě několika jednoduchých WWW

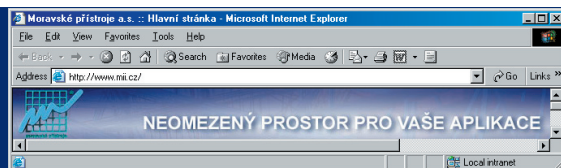
stránek nebývají dostupné již žádné další služby. Jinou možností je použít řídicí systém kompatibilní se standardem PC. I v této oblasti se však nasazení webového serveru prosazuje poměrně pomalu, přináší totiž pro tvůrce aplikace hned několik zásadních překážek:

- Webový server většinou není přímo součástí prostředí, v němž je vizualizace vyvíjena, proto se používají webové servery jiných výrobců. Tak přibývají problémy s přenosem dat (nejčastěji prostřednictvím databázového rozhraní). Jednak je tento přenos často pouze jednosměrný, ale navíc jsou stejná data zcela zbytečně ukládána v různém formátu na dvou místech (jednou pro archivaci a podruhé pro server). Tvůrce aplikace rovněž musí

zvládat používání několika různých vývojových prostředí a nástrojů pro vlastní řízení, vizualizaci a pro webový server.

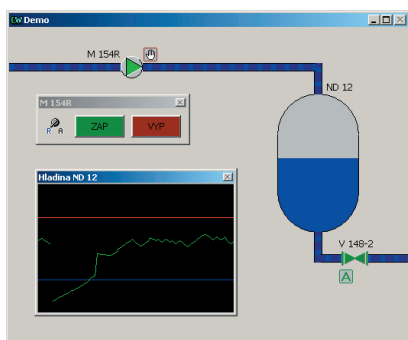
- Pokud je webový server součástí používaného vývojového prostředí, nemá uživatel-programátor stále vyhráno. Většinou je server připojen jen jako „modul“, a i když je komunikace se zbytkem aplikace zajištěná, její definice není triviální a většinou vyžaduje úpravu HTML dokumentů a psaní skriptů přímo ve zdrojovém tvaru.

Všechny výše zmíněné problémy odstraňuje nová verze rozšířeného systému **Control Web**. První novinkou je rozšíření vlastností zabudovaného webového serveru. Nový server umí vytvářet HTML stránky zcela dynamicky, bez nutnosti vytvářet soubory na disku. Byl zdokonalen systém vyrovnávací paměti a přibýly další možnosti volání metod



virtuálních přístrojů ve struktuře aplikace systému **Control Web**.

Další novinkou je průvodce pro vytvoření webového rozhraní aplikace. Uživatel systému **Control Web** tak může během několika sekund, bez znalosti HTML a webového serveru, získat webovou aplikaci, která nejenom že zobrazuje aktuální stav řízené technologie, ale může i nastavovat vybrané parametry.



Právě nutnost zapisovat identifikátory datových elementů přímo do textu HTML dokumentů a správná parametrizace webového serveru většinu uživatelů od používání této technologie zcela odradila.

Jak vlastně průvodce vytvářející webové rozhraní pracuje? V prvním kroku je vytvořena základní struktura aplikace. Uživatel vybere části zdrojové aplikace v prostředí **Control Web**, které si přeje zobrazovat. V dalším kroku zvolí ovládací prvky, které budou umístěny do webové aplikace a umožní změnu

parametrů nebo ovládní serverové aplikace (tedy aplikace běžící v systému **Control Web**).

Průvodce jednak modifikuje strukturu zdrojové aplikace, vkládá webový server a vytváří skripty. Dále pak vytvoří sadu HTML dokumentů na disku. Celý postup generování je možné několikrát opakovat a vytvořit tak víc různých rozhraní. V takovém případě může být ve výsledné aplikaci i několik webových serverů (každý bude pracovat na jiném portu).

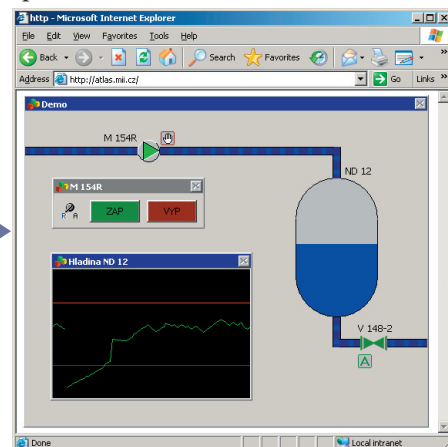


Průvodce dále umožňuje nastavit různé úrovně přístupových práv prohlížení a změny parametrů ve výsledné webové aplikaci.

Generátor ve výsledné aplikaci zachová vzhled zdrojové aplikace. Přístroje umístěné v okně vypadají stejně i v HTML aplikaci, stejně jako po spuštění aplikace vytvořené v prostředí **Control Web** lze přepínat panely nebo jimi pohybovat po pracovní ploše. Vše ovšem probíhá například v jednom Internet Exploreru.

Celý postup si můžeme ukázat na jednoduchém příkladu. Pro zjednodušení použijeme velmi malou aplikaci. Budeme ovládat čerpadlo pro doplňování vody

do zásobníku a zobrazovat aktuální hladinu v zásobníku. Aplikaci ukazuje obrázek 1. Nyní spustíme generátor. Zvolíme možnost „Vložit všechny prvky“, vybereme ovládací prvky, které bude možno nastavovat z webové aplikace a stiskem tlačítka „Dokončit“ webové rozhraní aplikace vytvoříme. Nyní stačí aplikaci spustit, otevřít webového klienta, zadat



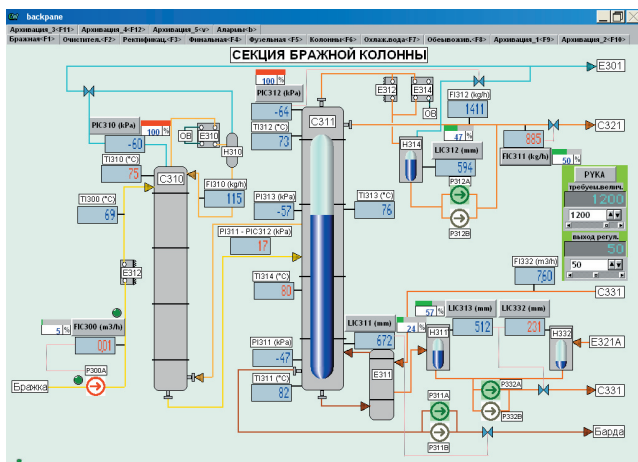
jméno počítače nebo IP adresu a vidíme výsledek generování. Díky technologii DHTML může výsledná aplikace vypadat téměř shodně s původní aplikací.

Jak je vidět, celý postup je velice jednoduchý a především odpadá nutnost nastavovat webový server, vytvářet HTML dokumenty nebo psát skripty. Jednodušeji již to nejde - stačí několik kliknutí myši a webová aplikace je hotova.

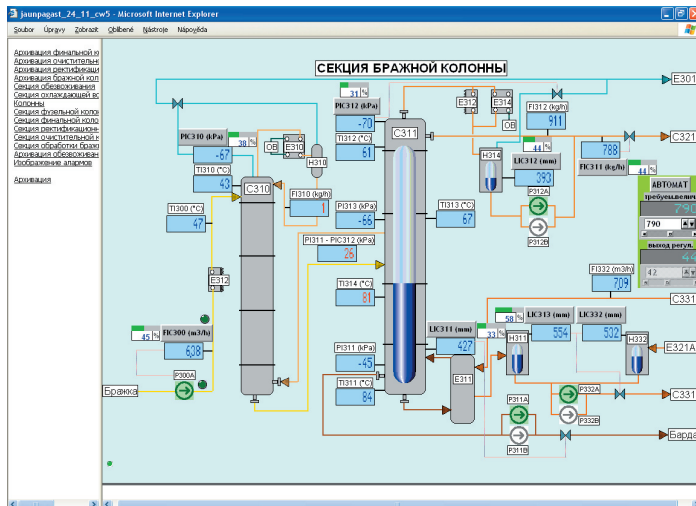
Ukázka využití webového přístupu k aplikačnímu programu

Aplikace v prostředí systému **Control Web** běží v nepřetržitých provozech ve spoustě zemí různě po světě. A často je velmi užitečné, když dodavatel má systém pod kontrolou, může na přání zákazníka software rozšiřovat a udržovat. Jenže cena benzínu, letenek a především času odborníků se stále zvyšuje. Jediné, co cestuje stále rychleji a levněji jsou data v celosvětovém Internetu. A řešení je tady - stačí vybavit aplikační program webovým rozhraním a již je lhostejno, pracuje-li spravovaný systém v sousední kanceláři nebo na druhé straně světa.

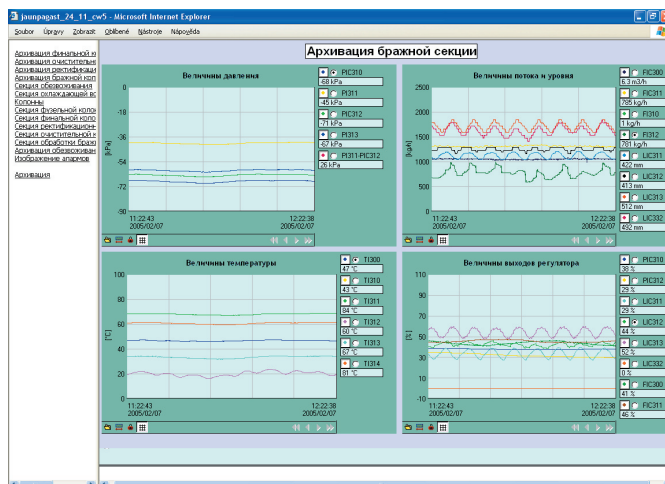
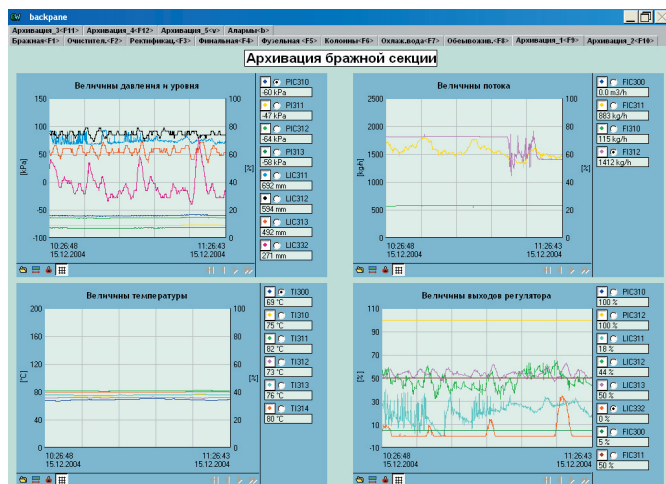
Řízení lihovaru na výrobu potravinářského lihu v Jaunpagasts (Lotyšská republika)



Dohled nad záparovou kolonou lokálně a dálkově přes Internet

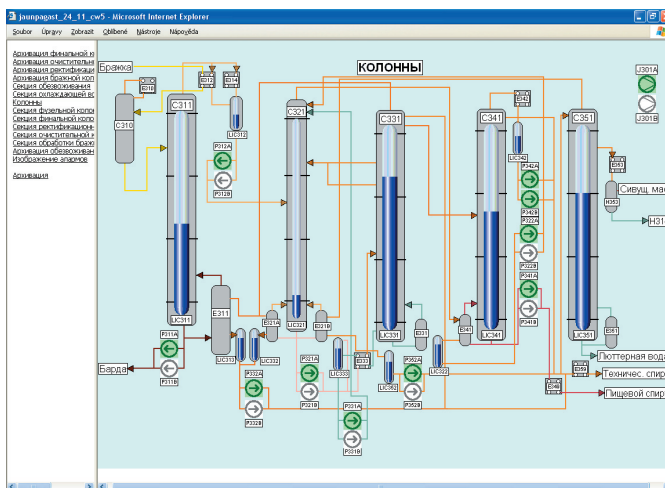
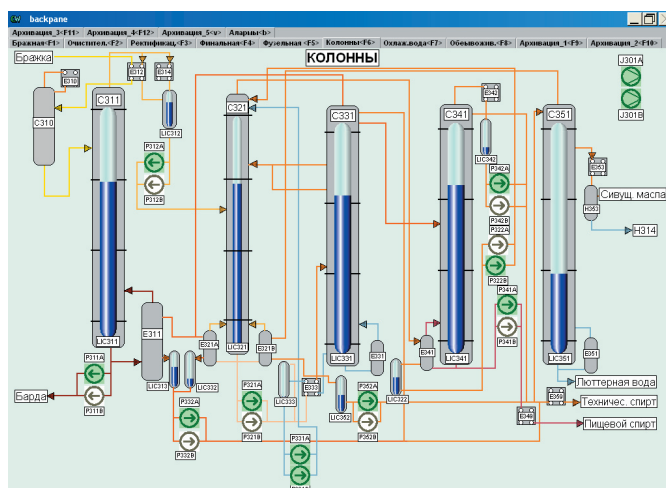


V lihovaru v Jaunpagasts realizovala VUCHZ, a. s. dodávku technologie vícetlaké destilace a následného odvodnění lihu s výslednou kapacitou 200 hl abs. alkoholu za den. Výstupem z destilační části je lih potravinářské kvality s koncentrací min. 96 obj.%, který se následně odvodňuje na lih pro farmaceutický průmysl s koncentrací min. 99,8 obj.%.



Ido archivů dat lze nahlížet nejen z lokálního operátorského pracoviště, ale i dálkově po Internetu

Technologický proces je řízen pěti automaty KS98+, každý se vstupním blokem RM200 na sběrnici CAN Bus. První automat řídí záparovou kolonu, druhý destilační kolonu, třetí čisticí a dokapovou kolonu, čtvrtý finální kolonu a pátý odvodňovací jednotku. Všechny automaty jsou propojeny sériovým kanálem RS485 s vizualizačním počítačem, který používá k vizualizaci získaných údajů program **Control Web**. Tento počítač je operátorskou stanicí, z níž se provádí řízení technologického procesu, zejména nastavování žádaných hodnot jednotlivých regulátorů a případně přechody mezi ručním a automatickým režimem jednotlivých regulačních smyček. Ovládání jednotlivých regulátorů z čelních panelů automatů je v případě normálního provozu zablokováno. V případě výpadku HW nebo SW vizualizačního počítače nebo výpadku jeho komunikace s automaty pokračuje řízení technologie bez přerušení, přičemž automaty se po výpadku přepínají do režimu ovládání regulačních smyček z čelních panelů rozvaděčů.



Destilační kolony online

Vizualizační počítač je zároveň webovým serverem, umožňujícím dálkovou kontrolu stavu technologie pomocí internetového prohlížeče.

System realizovala firma:

VUCHZ, a. s. vuchz@vuchz.cz
 Křižíkova 70 612 00 Brno
 Tel.: +420 - 541 633 111
 Fax: +420 - 541 211 181

Nezapomeňme na rizika

Internet je velmi užitečným, ale současně i docela nebezpečným prostředím. Jakmile je nějaký počítač dostupný jako server na veřejném Internetu, je v několikasekundových intervalech trvale vystavován útokům. Ne, že by stále tolik hackerů nemělo nic lepšího na práci, ale masivně a nepřetržitě se pokoušejí útočit roboty, které hledají nezazápatované servery, do kterých lze nějakým známým

postupem (nejčastěji buffer overrun) vniknout.

Proto je nutno mít vždy nainstalovány aktuální záplaty (operačního systému, případně http serveru, databázového systému atd.). Je také vhodné mít server za účinným firewallem. Těmi nejsilnějšími obrannými mechanismy však disponuje sám aplikační program.

Většina útoků je cílena na nejrozšířenější servery. Zde je značnou výhodou http serveru systému **Control Web**, že veškeré útoky na slabé stránky např. serverů Apache nebo Microsoft IIS se na **Control Web** „nechtají“. Aplikace v systému **Control Web** má také řadu možností, jak velmi účinně zabránit průniku nepovolané osoby. Je třeba tyto možnosti znát a využívat jich.

Technologie ActiveX

Technologie ActiveX se vyvíjela postupně během vcelku dlouhého časového období. Její základy byly položeny již v šestnáctibitových operačních systémech MS Windows; a to je skutečně dávná doba. Tehdejší motivace ke vzniku něčeho, co se následně přetavilo až do podoby moderního ActiveX, byla v zásadě shodná s dnešními základními požadavky na objektové komponentové technologie.

Předchůdci ActiveX měli za cíl vyřešit dva okruhy problémů:

- jak zajistit, aby se uvnitř programového systému dokázaly domluvit jeho jednotlivé části bez toho, aby se tyto části přesně (binárně) znaly. Vzájemná přesná znalost částí systému totiž znamená, že celý systém musí být vyroben a sestaven jako jeden monolit (i když je třeba rozdělen do více programových souborů). Monolitičnost je překážka údržbě a dalšímu vývoji systému; monolitičnost je také překážka spolupráci programů různých výrobců.
- jak zajistit, aby se určitý programový systém (třeba vývojový nástroj) dal snadno doplňovat a rozšiřovat o nové funkce, o nové vlastnosti a o funkční kousky programů naprosto cizích autorů. Jak sloučit programovou část kreslicí grafy s programovou částí, která dokáže pomocí měřicí karty sledovat teplotu?

Ukázalo se, že oba okruhy problémů vedou na totéž: jak vymyslet a standardizovat spojovací vrstvu programových částí —

dnes rozhraní (interface) — které jediné se následně bude pro spolupráci a výměnu dat programových částí používat. Je-li určen způsob, jak se rozhraní vytváří a používá, je snadné splnit oba motivační cíle:

- programy různých výrobců se domluví, budou-li mít pro stejný účel totéž rozhraní
- doplňky a rozšíření lze použít, jsou-li známa jejich rozhraní poskytující určitou funkci

Princip použití programového rozhraní je v dnešních informačních technologiích naprosto zásadní. Platí to beze zbytku

rovněž pro hlavní objektovou technologii operačních systémů Windows, kterou je Component Object Model (COM). COM sám o sobě je samozřejmě pouze prázdný rámec, který dokáže poskytnout rozhraní objektů a který dokáže zaručit jejich propojitelnost. Bez dalšího určení, jak a čím bude tento rámec naplněn, bez určení pravidel a naplánování rozhraní, nelze COM rozumně používat (a stejně tak ani žádnou jinou komponentovou technologii).

Objektový model technologie ActiveX je COM. To je hlavní a základní vlastnost ActiveX, která prakticky kompletně vysvětluje a popisuje, co a jak lze pomocí ActiveX vyřešit. Současně je tím však opět řečeno, že bez stanovení pravidel a naplánování rozhraní je ActiveX stejně neurčitý, jako samotný COM.

Než se však o některých existujících pravidlech a blocích rozhraní zmíníme, je nutné učinit odbočku k Internetu. Technologie ActiveX má totiž důležitou vlastnost, která přímo s rozhraními a objekty nespojuje. Při skládání jednotlivých objektových komponent nemusejí být všechny přímo dostupné (tj. například na místním disku), mohou být uchovány na cizích místech. Takové cizí místo může být například u výrobce komponenty, u poskytovatele určité služby (například u banky v případě elektronického bankovníctví) nebo to může být jen prosté skladiště užitečných komponent. ActiveX řeší,

pomocí Internetu, jak takovéto vzdálené komponenty zpřístupnit natolik, aby je bylo možné jako komponenty použít. Důležité současně je, že tato vlastnost technologie ActiveX vůbec nijak neomezuje a nezakazuje čisté místní použití komponent.

Pojďme zpět k objektovým rozhraním. Stanovením pravidel a naplánováním rozhraní (skupiny rozhraní) vždy vznikne řešení pro nějaký konkrétní účel. V zásadě i tento samotný účel může být různě konkrétní, takže vznikají řešení různě obecná, různě rozšiřitelná a různě upravitelná; vznikají různé objektové a komponentové standardy. Příklad několika z nich bude nejnázornější:

OPC (OLE for Process Control)

Standard a technologie pro unifikovaný

přístup k řízení, měření a regulaci průmyslových procesů. Sada rozhraní rozoznává OPC server a OPC klient.

OLE (Object Linking and Embedding)

Standard a technologie pro zařazování dokumentů dovnitř jiných dokumentů (příkladem je tabulka MS Excel vložená dovnitř dokumentu MS Word); uvnitř kancelářského balíku MS Office je OLE zcela skryté a automatické.

ActiveX controls

Standard především pro komponenty uživatelského rozhraní — pomocí ActiveX controls se sestavují aplikace, dialogová okna apod. Každý ActiveX control musí být vidět (musí mít rozhraní pro uživatelské ovládání



a pro operace související se vzhledem), musí mít rozhraní pro programové ovládání, a pokud to má pro konkrétní ActiveX control smysl, může mít rozhraní pro poskytování svých událostí.

ActiveX automation object

Standard pro COM komponenty, které nejsou na obrazovce vidět. Každý ActiveX automation prvek musí mít rozhraní pro programové ovládání, a pokud to má pro konkrétní ActiveX automation prvek smysl, může mít rozhraní pro poskytování svých událostí.

Je vcelku zřejmé, že ActiveX automation prvek je ActiveX control zbavený rozhraní a funkčnosti související s uživatelským rozhraním. Obě varianty technologie ActiveX se často směšují, rozdíl je totiž velmi malý. Řekne-li se ActiveX, může se jednat o obojí druh komponenty.

Všechny komponenty vystavené v objektovém modelu COM jsou ve všech nástrojích, které této technologii rozumějí, dostupné. Teoreticky by tak mělo být možné v každém nástroji libovolně spojovat funkce jakýchkoli standardů. Prakticky to není vždy možné. Platí zpravidla, že nástroje určené k programování (Visual Basic) dokáží COM využít skutečně obecně, zatímco jiné nástroje využívající COM využívají jen některý objektový standard (třeba OPC, nebo ActiveX).

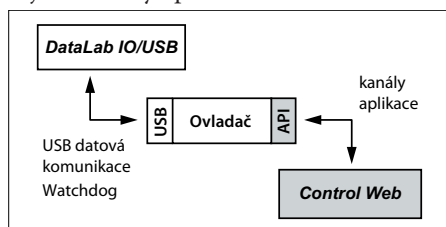
Přestože je variabilita komponentových standardů velmi obecná a bohatá, po-

užívá se v praxi pro rozšiřování funkcí a spojování komponent téměř výhradně ActiveX. Pravděpodobně proto, že bloky rozhraní ActiveX a pravidla pro jejich používání nejsou v ActiveX nijak komplikovaná a že ActiveX lze velmi snadno rozšiřovat a používat v prostředí Internetu.

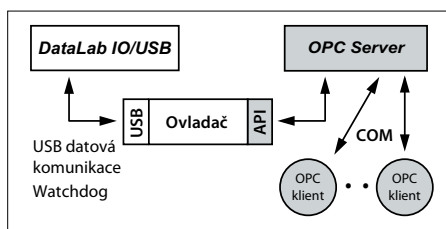
Ovladače jsou prvky ActiveX

Skutečnost, že technologie ActiveX je pro rozšiřování vlastností a komunikaci s okolními programy základní a téměř výhradní, využívají i ovladače jednotek průmyslového měřicího a řídicího systému DataLab. Všechny ovladače jednotek DataLab tak lze použít jedním ze třech způsobů:

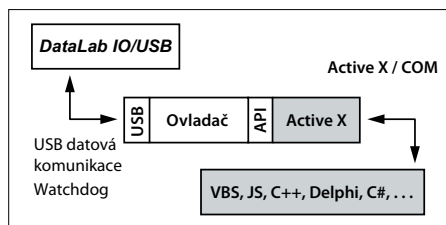
- V aplikaci **Control Web**. Zde se jedná o nativní způsob využití ovladače, kdy jsou data vstupů a výstupů dostupná jako kanály aplikace.



- V **Univerzálním OPC serveru**. Tento způsob využití ovladače předpokládá instalaci produktu Univerzální OPC server. Data vstupů a výstupů, případně komunikace jednotek jsou dostupné prostřednictvím OPC serveru jakémukoli OPC klientovi.



- Jako **ActiveX automation prvek**. Způsob využití, který nepředpokládá instalaci žádného produktu (samozřejmě kromě toho, v němž má být ovladač jako ActiveX prvek použit). Zde samotná knihovna ovladače nabízí všem COM klientům svá ActiveX rozhraní.



Pro široké použití a univerzální zapojení jednotek **DataLab** do všech cizích produktů je tento způsob využití ovladače

nejjednodušší a automaticky nejpřímější.

Použije-li se ovladač jako ActiveX automation prvek, není kromě samotné knihovny ('DLL' soubor) ovladače třeba nic více dodávat. Co je však vždy třeba, je registrace ovladače jako ActiveX prvku. Má-li totiž být ActiveX prvek cizím programovým systémům dostupný, musí o něm vědět operační systém — technicky to nutné není, protože i jen s pomocí DLL lze COM využít; prakticky však všechny produkty používající ActiveX získávají data o komponentách jen z operačního systému.

Nejjednodušší způsob registrace je s využitím programu 'regsvr32.exe' z příkazové řádky operačního systému — příkazový řádek operačního systému je program 'cmd.exe'; spustit tento program lze například příkazem Spustit... nabídky Start operačního systému. Příklad registrace ovladače **DataLab IO/USB**:

```
c:\Documents>cd \Directory\With\DLL
c:\Directory\With\DLL>regsvr32 dldrv.dll
c:\Directory\With\DLL>
```

Opačná operace, zrušení registrace, se provede velmi podobně. Pouze se volání programu regsvr32 rozšíří o volbu /u (uninstall):

```
c:\Documents>cd \Directory\With\DLL
c:\Directory\With\DLL>regsvr32 /u dldrv.dll
c:\Directory\With\DLL>
```

Je evidentní, že pro jiné ovladače systému **DataLab** (například **DataLab IF/EIB**) je třeba registrovat jinou DLL (například 'dleibdrv.dll').

Sama registrace, tak jak je zde popsána, nemusí být nutná v případě, kdy již ovladač jako ActiveX registroval někdo jiný (například instalátor ovladače). Tak či tak se případnou druhou registrací nic nepokazí a lze ji bez obav provádět i opakovaně.

Okamžikem registrace komponenty ji lze použít ve všech programech rozumě-

jících ActiveX. Záleží pak na konkrétním programu a jeho zvyklostech (či dokonalosti), jak se k ActiveX prvku ovladačů propracují. Pravděpodobně však většina programů bude požadovat buď unikátní identifikátor ActiveX prvku (GUID), nebo symbolické jméno ActiveX prvku (ProgId). Například pro ovladač **DataLab IF/EIB** je tato dvojice GUID a jména následující: {5D9DAB98-D82D-4090-984A-C017768BE03C} a ControlWeb.DrvAxDllIfEib. Navíc většina nástrojů pro výběr komponenty nabízí nějaký vizuální nástroj (dialogové okno apod.), takže výběr správného ActiveX prvku skutečně není složitý.

Příklady použití

Ovladače systému Control Web lze jako ActiveX použít na mnoha místech. Všechna možná místa použití proto nelze dostatečně dobře podchytit. Protože je však ActiveX standard, budou i nepodchycená místa v principu podobná uvedeným příkladům.

Každé použití ovladače jako ActiveX předpokládá, že pro konkrétní jednotku systému **DataLab** bude existovat správně sestavený konfigurační (parametrický) soubor. Tento konfigurační soubor lze umístit kamkoli, protože při rozběhu ActiveX prvku může být cesta k němu uvedena kompletní.

Obsah konfiguračního souboru se liší ovladač od ovladače, a přestože soubory mají formálně stejný zápis (dělí se na sekce a klíče), je jejich zápis nutné vždy konzultovat s dokumentací ovladače (dokumentace je součástí každého ovladače). Jako základ konfiguračního souboru lze vždy vzít nějaký existující (například ukázkový) konfigurační soubor.

Ukázka kódu VBScript pro komunikaci s **DataLab IO/USB** uvnitř HTML dokumentu

```
<html>
<head>
  <title>DataLab IO ActiveX sample</title>
  <object
    id="DataLab"
    classid="clsid:45D4DCAF-DCE7-4484-9321-8E12C7B7E8DB">
</head>
<script language="VBScript">
sub Init()
  window.SetInterval "OnInterval", 10
  DataLab.LoadPARFile "SOMEWHERE\d1.par", ""
end sub
sub OnInterval()
  ReadAnalog()
end sub
sub ReadAnalog()
  DataLab.MarkInput 500
  DataLab.ReadInputs()
end sub
```

```

sub DataLab_OnInputRead( CS, Index, EC, Value )
    AnalogBar.style.width = CLng( Value ) / 32768 * 500
end sub

```

```

Init()
</script><body>

```

```

<p>Read analog value:
<span
    id="AnalogBar"
    style="background: red;"></span></p>
</body></html>

```

Uvedený krátký HTML dokument pomocí jednotky DataLab IO/USB načítá údaj z analogového vstupu a zobrazuje jej jako proměnlivě široký červený pásek. V ukázce je v místě volání procedury DataLab.LoadPARFile použit konfigurační soubor 'SOMEWHERE\dl.par', který, samozřejmě, musí být nahrazen cestou se správným umístěním souboru (cestou na místním disku). V ukázce je dále použito číslo kanálu 500 (DataLab.MarkInput 500), toto číslo musí odpovídat definovanému číslu kanálu v konfiguračním souboru. Bude-li v tomto souboru vstupu přiřazeno jiné číslo, bude i volání MarkInput obsahovat jiné číslo.

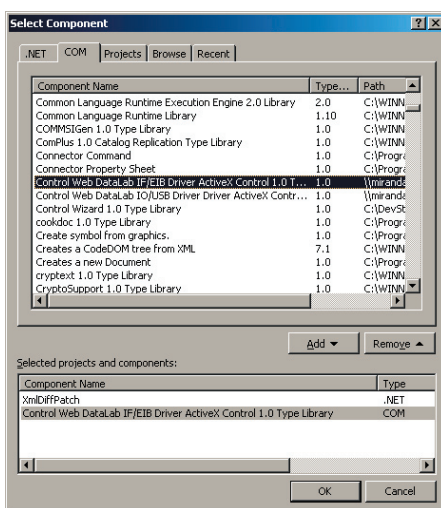
Použití ActiveX v MSVisual Studio

ActiveX komponentu pro **DataLab IO/USB** je možné používat uvnitř vývojového prostředí stejně jako každou jinou komponentu. V zásadě je nutné učinit následující kroky:

- zařadit komponentu do projektu
- definovat komponentu, případně včetně jejích událostí
- použít komponenty v programu

Zařazení komponenty do projektu

Komponenta se zařadí po výběru ze záložky COM. Tato záložka zobrazuje COM komponenty známé operačnímu systému, přičemž se nemusí vždy jednat o ActiveX prvky — přestože jich velká většina je právě taková. Obrázek ukazuje nalezený ActiveX prvek ovladače DataLab IF/EIB již po té, co byl do projektu zařazen (tlačítkem Add).



Vybraný a přidáný ActiveX prvek ovladače DataLab IF/EIB

Definice objektu a připojení jeho událostí

Definice komponenty je odlišná podle jazyka a platformy, která se používá. Tedy například, takhle nějak může vypadat vytvoření objektu v C#:

```

...
private dldrv_ax.dldrv_ax DataLab = null;
...
DataLab = new dldrv_ax.dldrv_ax();
...

```

nebo ve jazyku Visual Basic:

```

...
Private DataLab As dldrv_ax.dldrv_ax = null
...
DataLab = New dldrv_ax.dldrv_ax()
...

```

Stejně tak se velmi může lišit spojení objektu s jeho událostmi — od jednoduchého zápisu události, až po komplikovanější tvorbu spojovacích objektů. To vše je nicméně dobře v každém vývojovém nástroji popsáno včetně průvodců a dokumentace.

Vlastní použití komponenty

Použití komponenty je již potom jednoduché, tak jak to je koneckonců uvedeno

v ukázce skriptu v HTML dokumentu. Princip práce ActiveX prvku je podrobně popsán v dokumentaci, je však velmi jednoduchý, takže nic nebrání popsat jej stručně v následující sekci.

Princip práce ActiveX prvků

Operace čtení (neboli přenos dat z technologie či hardware do počítače — programu) i operace zápisu (opačný přenos dat z počítače — programu do technologie či hardware) vždy probíhají ve třech krocích:

- shromáždění a připravení seznamu dat, která se mají číst nebo zapsat
- zahájení čtení nebo zápisu a průběh těchto operací
- zpracování informací o dokončení čtení nebo zápisu

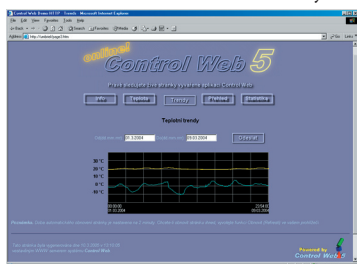
Tyto tři kroky komunikace probíhají dávkově. Po shromáždění požadavků se zahájí komunikace a dokud tato probíhající komunikace neskončí není možné zahájit komunikaci novou. Neboli, dokud nejsou dokončeny (přečteny nebo zapsány) všechny komunikace v dávce, nelze zahájit dávku novou; dovnitř jedné dávky nelze vsunout jinou. Avšak — jednotlivá čtení i zápisy lze i uvnitř běžící dávky připravovat pro dávku novou.

Všechny jednotlivé měřené či zapisované body jsou označeny číslem (indexem), pomocí kterých se identifikují. V principu je možné pro identifikaci použít jakákoli čísla, technicky jsou však všechna možná čísla omezena aktuálním nastavením, které je přesně zapsáno v konfiguračním souboru (soubor má obvyčejně příponu 'PAR').

Procedury ActiveX komponenty lze rozdělit do třech skupin — procedury pro nastavení a řízení, procedury pro zahajování komunikace a procedury pro zpracování výsledků komunikace. Procedury pro nastavení a řízení se typicky použijí jednou, procedury pro zahajování a řízení komunikace pak násobně tak, jak je třeba data zapisovat nebo číst.

Na závěr

Co říci na závěr? Snad jen tolik, že všechny dostupné informace, které jsou v tomto textu jen zmíněny, je možné velmi pohodlně získat buď na CD s ovladači dodávaném s produkty **DataLab**, anebo rovněž velmi pohodlně, na internetových stránkách společnosti Moravské přístroje — <http://www.mii.cz>.



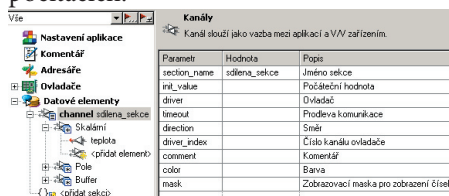
Datové sekce - silný nástroj pro distribuovaná data

Nedělejme síťové distribuované aplikace postaru! Mechanismy výměny a správy dat rozptýlených v počítačových sítích, které se dříve musely v aplikacích docela promyšleně programovat, jsou nyní okamžitě a s veškerým komfortem k dispozici.

Control Web 5 přináší nové možnosti vpro tvorbu distribuovaných aplikací. Hlavní motivací při návrhu nového systému byla snaha ušetřit práci při vývoji rozsáhlých síťových aplikací.

Tento článek na jednoduchém příkladu ukazuje, že vytváření rozsáhlých distribuovaných aplikací je v prostředí **Control Web 5** velice jednoduché a zvládne ho skutečně každý.

V systému **Control Web 5** jsou nyní všechny datové elementy (konstanty, proměnné, kanály ...) umístěny v tzv. **datových sekcích**. Datová sekce je základním prvkem, který je možno sdílet mezi aplikacemi běžícími na různých počítačích.



Parametr	Hodnota	Popis
section_name	sdilena_sekce	Jméno sekce
init_value		Počáteční hodnota
driver		Ovladač
direction		Prodeva komunikace
lineout		Směr
driver_index		Číslo kanálu ovladače
comment		Komentář
color		Barva
mask		Zobrazovací maska pro zobrazení čísel

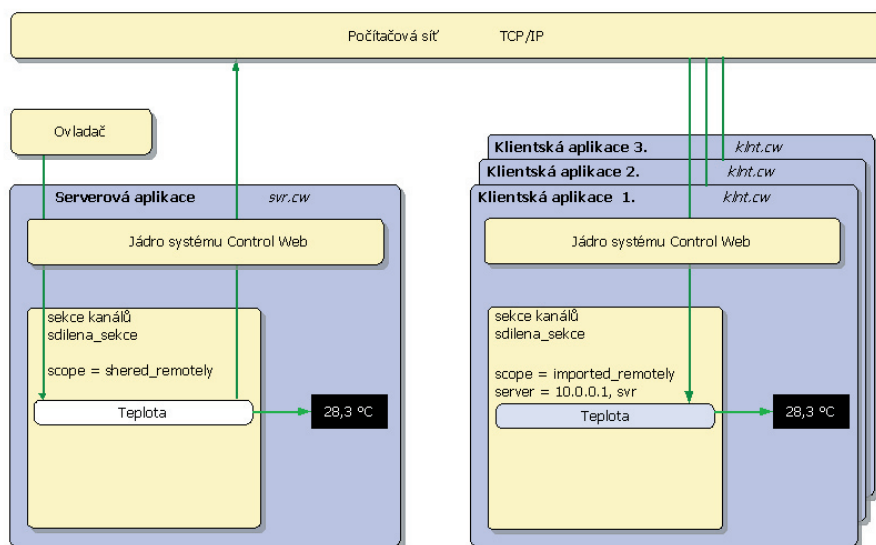
Distribuovaná aplikace v CW - sdílené sekce kanálů

V systému **Control Web** vytvoříme novou aplikaci. Toto bude serverová aplikace, která bude poskytovat data dalším aplikacím. V aplikaci vytvoříme sekci kanálů a do ní vložíme kanál teplota, který bude prostřednictvím ovladače získávat teplotu z technologie (například komunikací s PLC). Sekci pojmenujeme **sdilena_sekce**. Takto vytvořený kanál je možné používat v celé lokální aplikaci.

Nyní si ukážeme, jak můžeme k tomuto kanálu přistupovat z dalších aplikací, ať už z lokálního počítače nebo z libovolného počítače lokální sítě.

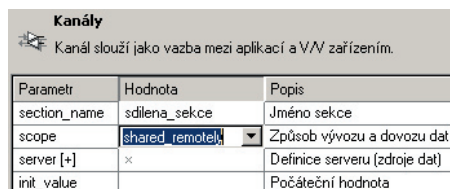
Prvním krokem je zpřístupnění sekce kanálů v serverové aplikaci. Sekci nastavíme jako sdílenou – nabízenou pro ostatní aplikace. V serverové aplikaci se přepneme do datových inspektorů, vybereme sekci kanálů a najdeme parametr **scope**. Parametr **scope** je klíčovým při vytváření distribuovaných aplikací. V serverové aplikaci, v sekci kanálů nastavíme parametr **scope** na hodnotu **shared_remotely**. Tato sekce bude od tohoto okamžiku nabízená všem aplikacím systému **Control Web** běžícím na

jakémkoliv počítači, dosažitelném prostřednictvím počítačové sítě.



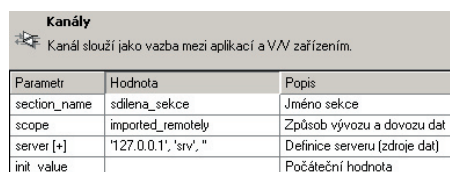
V druhém kroku vytvoříme klientskou aplikaci, v níž zpřístupníme serverovou sekci kanálů. Vytvoříme novou aplikaci. Vložíme do ní novou datovou sekci

můžeme zadat IP adresu lokálního počítače 127.0.0.1



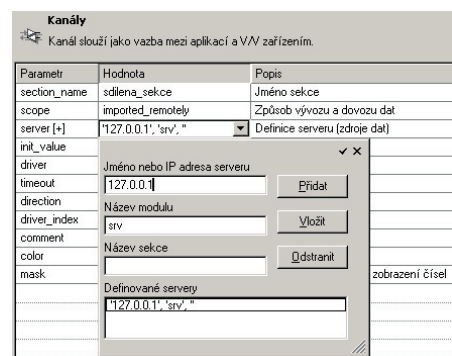
Parametr	Hodnota	Popis
section_name	sdilena_sekce	Jméno sekce
scope	shared_remotely	Způsob vývozu a dovozu dat
server [+]	x	Definice serveru (zdroje dat)
init_value		Počáteční hodnota

kanálů. Tuto sekci pojmenujeme také **sdilena_sekce**. Nyní opět nastavíme parametr **scope**. U klientské sekce nastavíme **scope** na hodnotu **imported_remotely** (vzdáleně dovážená sekce). Nyní bude **Control Web** datové elementy v této sekci dovážet ze vzdáleného počítače. Musíme ještě určit, odkud budou datové elementy dováženy. Musíme tedy



Parametr	Hodnota	Popis
section_name	sdilena_sekce	Jméno sekce
scope	imported_remotely	Způsob vývozu a dovozu dat
server [+]	'127.0.0.1', 'srv', ''	Definice serveru (zdroje dat)
init_value		Počáteční hodnota

určit počítač, na němž poběží serverová aplikace, jméno serverové aplikace a jméno sekce, z níž budou dovezené datové elementy. Toto všechno nastavíme parametrem **server**. Tento parametr nastavíme opět v datových inspektorech. Pro úpravu parametru **server** nabízí **Control Web** jednoduchý editor, v němž musíme vyplnit následující hodnoty:



Parametr	Hodnota	Popis
section_name	sdilena_sekce	Jméno sekce
scope	imported_remotely	Způsob vývozu a dovozu dat
server [+]	'127.0.0.1', 'srv', ''	Definice serveru (zdroje dat)
init_value		Počáteční hodnota
driver		Jméno nebo IP adresa serveru
timeout	[127.0.0.1]	
direction		Název modulu
driver_index		Číslo kanálu ovladače
comment		Komentář
color		Barva
mask		Zobrazovací maska pro zobrazení čísel

-**Název modulu** – jméno serverové aplikace (jméno zadejte bez přípony cw)

-**Název sekce** – jméno sekce kanálů v serverové aplikaci. Protože v naší klientské aplikaci je sekce pojmenovaná stejně jako v serverové aplikaci, nemusíme tento parametr vyplňovat.

Sekce kanálů v klientské aplikaci je definována jako prázdná – neobsahuje definici žádných datových elementů. Fyzicky při spuštění aplikace však sekce kanály obsahuje, seznam kanálů je načten ze serverové aplikace.

Nyní můžeme v obou aplikacích číst teplotu, například do přístroje meter nastavíme vstup (parametr **input**) **sdilena_sekce.teplota**. V serverové aplikaci **Control Web** přečte hodnotu prostřednictvím ovladače. V klientské aplikaci musí **Control Web** přečíst

hodnotu ze serverové aplikace, která ji přečte z ovladače. To však dělá jádro systému **Control Web**. Tvůrce aplikace nemusí vědět, jestli je kanál lokální nebo přenášený z jiného počítače. Vždy pracuje s kanálem **teplota** v sekci **sdilena_sekce**. Práce s kanálem je naprosto stejná v serverové i klientské aplikaci.

Takto vytvořenou klientskou aplikaci můžeme spustit na libovolném počítači naší lokální sítě. Můžeme ji spustit v jeden okamžik na libovolném počtu počítačů nebo na jednom počítači víckrát. Všechny aplikace budou zobrazovat hodnotu **teplota**. Pokud budou počítače naší lokální sítě přístupné z Internetu, můžeme klientskou aplikaci spustit na počítači kdekoli na Internetu – vždy bude správně zobrazovat naši teplotu.

V uvedeném postupu byly záměrně definovány pouze základní parametry, aby ukázka byla co nejjednodušší a nejnázornější. Při praktickém použití jsou možnosti systému **Control Web** mnohem rozsáhlejší. Je například možné v serverové aplikaci nastavit, kteří klienti se mohou ke sdílené sekci připojit nebo kolik klientů může být v jeden okamžik připojených (parametr **client_access**).

Další možností je nadefinovat v klientské aplikaci v parametru sekce **server** více počítačů, na nichž může běžet serverová aplikace. Potom stačí, aby aplikace běžela alespoň na jednom z nich a **Control Web** se k němu automaticky připojí. Pokud je serverová aplikace zastavena, najde si klientský **Control Web** jiný spuštěný server (pokud takový existuje). To vše dělá **Control Web** zcela automaticky, aniž to uživatel za běhu aplikace postřehne. Pokud chce mít tvůrce aplikace kontrolu nad výběrem serverů, nabízí **Control Web** celou řadu nativních a událostních procedur.

Obrovskou výhodou při vývoji aplikací v systému **Control Web** je možnost testovat klientské aplikace na lokálním počítači a po dokončení jej bezpečně přenést na jinou pracovní stanici. To v praxi značně urychlí tvorbu aplikací.

Archivace dat

Nyní naši aplikaci rozšíříme o možnost archivace změřené teploty. Změřenou teplotu budeme chtít archivovat jednou za minutu. Do serverové aplikace přidáme novou datovou sekci typu **archive**. Sekci pojmenujeme **archive**. Aby sekce archivovala jednou za minutu nastavíme parametr **timer** sekce na hodnotu 60 (parametr je v sekundách).

Parametr	Hodnota	Popis
section_name	archive	Jméno sekce
scope	shared_remotely	Způsob vývozu a dovozu dat
server [+]	x	Definice serveru (zdroje dat)
init_value		Počáteční hodnota
timer	60	Časovač aktivace
condition		Podmínka aktivace
timeout		Prodleva komunikace
comment		Komentář
color		Barva

Protože budeme chtít k archivu přistupovat i z klientské aplikace nastavíme parametr **scope** na hodnotu **shared_remotely**.

Nyní musíme do archivační sekce vložit novou položku pro archivaci. V seznamu datových elementů klikněte na přidat a zadejte jméno datového elementu **archiv_teplota**. Nyní zbývá nastavit, aby tento nový datový element archivoval hodnotu kanálu **teplota**. V datových inspektorech nastavte parametr **expression** datového elementu na **sdilena_sekce.teplota**.

Pokud nyní serverovou aplikaci spustíme bude **Control Web** automaticky každou minutu ukládat změřenou hodnotu teploty. Dále si ukážeme, jak je možné archivované hodnoty prohlížet.

Nejprve zpřístupníme archivní sekci. V klientské aplikaci vytvoříme novou archivní sekci, kterou pojmenujeme stejně jako v serverové aplikaci **archive**. Nastavíme opět parametr **scope** na hodnotu **imported_remotely** a parametr **server** jako v předchozím případě.

Posledním krokem je zobrazení historie teploty. Do klientské aplikace vložíme zobrazovač datových elementů přístroj **data_viewer**. V inspektoru přístroje nastavíme parametr **timer** na 60 sekund, přístroj bude obnovovat zobrazený

graf jednou za minutu. Dále v parametru **data_group** přidáme odkaz na archivovaný datový element z dovezené datové sekce: **archive.archive_teplota**.

Nyní můžeme spustit server a libovolný počet klientů. Ve všech klientských aplikacích uvidíme stejný graf průběhu teploty. Historická data jsou uložena u serverové aplikace, klientské aplikace data pouze zobrazují.

Alarmy

Další možností je rozšířit aplikace o sledování alarmů, například překročení určené meze měřené teploty. Postup je obdobný jako v předchozím případě. Do serverové aplikace vložíme novou datovou sekci typu **alarm**. Do klientské aplikace se přidá datová sekce a prohlížeč alarmů – přístroj **alarm_viewer**.

Potom jsou ve všech klientských aplikacích (nezávisle na tom, na kterém počítači běží) zobrazeny stejné alarmy. Je možno prohlížet jejich historii, potvrzovat vzniklé alarmy a podobně. **Control Web** zajistí, že na všech běžících aplikacích je vždy zobrazen stejný stav.

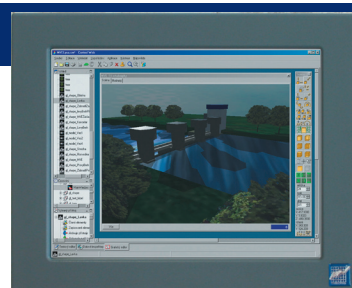
Synchronizace sekcí

Synchronizace sekcí (**scope = synchronized-shared-remotely**) umožňuje vytvořit více datových sekcí ve dvou či více aplikacích pracujících na různých počítačích, z nichž ale jen jedna je aktivní. To znamená, že jen jedna aplikace např. komunikuje s daným PLC či jiným hardware, ostatní aplikace jen synchronizují obsah datových elementů po síti. Jestliže aktivní aplikace z nějakého důvodu selže (např. zastaví se ventilátor na chladiči procesoru), aktivní roli převezme jiný počítač a systém běží bez přerušení dále.

Radek Seeman, seeman@mii.cz

Nová konstrukce průmyslových LCD monitorů

Průmyslové monitory **DataLab LCD** s dotykovými obrazovkami se těší poměrně značnému zájmu zákazníků. Proto jsme se rozhodli zlepšit jejich konstrukci. Nový monitor obsahuje méně vnitřní kabeláže, což přináší lepší kvalitu obrazu, je lépe větraný a pro jeho stavbu jsou použity kvalitnější a lépe zpracované materiály. Nosná konstrukce je tvořena monolitickým masivem z hliníkové slitiny. Důkladněji je vyřešeno také přední těsnění dotykové obrazovky. Ve variantě bez dotykové obrazovky je monitor vybaven čelním tvrzeným sklem.



Control Web a EIB v malém

Popisujeme-li jednotlivá nasazení systému **Control Web**, většinou se soustředíme na rozsáhlé a složité aplikace. Schopnost zpracovávat desetitisíce vstupně/výstupních signálů jistě dokumentuje schopnosti systému. Na druhé straně může být neméně důležitá také možnost použití prostředí **Control Web** i pro malé aplikace. Nízká cena průmyslových počítačů **DataLab** spolu s runtime systémem **Control Web** umožňuje ekonomické používání těchto moderních prostředků při automatizaci rodinných domů a bytů.

Instalační sběrnice EIB je v povědomí odborné veřejnosti známa převážně jako řešení, které se používá především ve velkém — v administrativních a správních budovách, sportovní halách či hotelích. Tento stav je pravděpodobně důsledkem dvou skutečností: jednak, velké stavby obvykle projektují renomované projekční kanceláře (které EIB rozumí a umějí ji použít) a jednak, velké stavby zpravidla financují velcí investoři, kteří si dobře uvědomují, že provoz budovy bude něco stát a že je nutné úsporný chod budov připravovat od samého počátku.

Taková velká řešení s EIB jsou samozřejmě známá. EIB má však potenciál i pro malá řešení, kdy se s velkou výhodou využijí jiné příhodné vlastnosti — například místo celkových energetických úspor maximální sdílení společných funkcí a využívání dostupných informací v mnoha kontextech (konec konců, sdílení informací je jedním z klíčových prvků standardu EIB).

Malým řešením může být například jediný rodinný dům, či dokonce byt. I v samostatném bytě je mnoho věcí, které má smysl integrovat do funkčního celku. Základní věci zvládne samotná EIB síť.

Co však již EIB nezvládne, jsou složitější logické operace, internetová komunikace, sestavování přehledových grafů, archivace údajů nebo GSM komunikace. Tedy, přesněji řečeno, určité z těchto úkolů pomocí EIB-nativních počítačů (řídících jednotek) vyřešit lze, ale zpravidla ne zcela volně a zpravidla ne najednou. Má proto smysl spojit EIB s obecným počítačem, který pomocí nějakého automatizačního systému, například systému **Control Web**, všechny popsané složitější operace uskuteční.

Co a proč má smysl v bytě pomoci EIB sledovat a řídit?

Vytápění

Většina bytů má více pokojů a většina pokojů není stále obsazena. Má proto smysl řídit vytápění tak, aby se v

místnostech topilo pouze, když v nich někdo je. Samozřejmě, za automatické lze považovat řízení topení podle denní doby.

Osvětlení

Svítit je možné různě. Obecně platná jsou hygienická pravidla — místnost by měla mít obecné centrální stropní světlo a místní specializovaná světla pro specializované činnosti (čtení, světlo u pracovního stolu apod.). Každý dobře ví, že různé úkony potřebují různé osvětlení; je dobré svítit jinak při sledování televize, jinak při posezení s přáteli a jinak při úklidu. Navíc podobně jako při vytápění má často smysl v místnosti svítit pouze, když v ní někdo je. A případně může mít smysl okamžitě po opuštění místnosti v ní zase zhasnout. A vůbec může být dobré některá světla zhasínat po určité době vždy.

Přístup venkovního světla

neboli ovládání žaluzií. Přes den je možné lamely naklápět automaticky (třeba podle aktuální polohy Slunce), večer je možné žaluzie automaticky stáhnout. Ne že by bylo složité pracovat se žaluziemi ručně, ale co se děje naprosto automaticky, funguje opakovaně lépe, než při lidských zásazích.

Zabezpečení

Jistotu, že v bytě nejsou nezvaní hosté, by rádi měli všichni. Standardní řešení dneška představuje samostatná vedení, samostatné senzory a samostatnou ústřednu. Řešení s pomocí EIB představuje jen využití informací z přístrojů, které jinak slouží i jiným účelům. Se zabezpečením souvisí i osvětlování, případně ovládání žaluzií: mnoho zlodějů před vniknutím nejprve sleduje, jestli je byt opuštěný. A pokud se bude večer svítit a přes den manipulovat žaluziemi, i když v bytě nikdo nebude, zloděj bude mít situaci těžší.

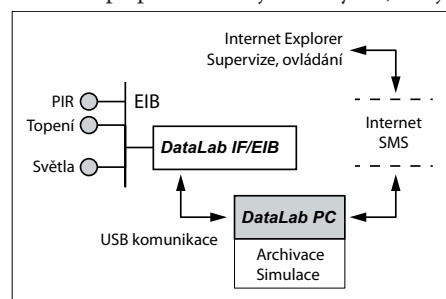
Komplikovanější úkony v nastíněných okruzích lze vyřešit oběma zmíněnými cestami — buď s využitím speciálního EIB prvku anebo s využitím řídicí aplikace systému **Control Web** (jako její jedné, typicky malé) části. Jako příklad

může sloužit simulované svícení; buď se do EIB osadí drahý samostatný prvek, který dokáže sledovat provoz světel a pak je opakovat; anebo se provoz světel uchová uvnitř aplikace systému **Control Web** a opakování (simulaci) v případě potřeby opět vyřeší **Control Web**. Neboli, shrnuto, vždy je možné potřebný díl rozšířených funkcí vytvořit samostatným jednoúčelovým prvkem, nebo je možné všechny tyto rozšířené funkce zařadit, za jedny peníze, do jediné programové aplikace.

Přenos funkcí do obecného programu přináší automaticky všechny výhody programového řešení — takové řešení je snadno udržovatelné, snadno rozšiřitelné a v případě potřeby i docela jednoduše zcela přepracovatelné. Kromě toho v sobě **Control Web** automaticky nabízí ohromné komunikační možnosti — přes internetový www server, či komunikaci s jinými systémy **Control Web** či komunikaci s jinými zařízeními obecně (GSM) apod.

Konkrétně

Výše popsané obecné principy a skutečnosti lze snadno demonstrovat na konkrétním případě existujícího bytu. Jaké je



v tomto bytě použito technické řešení?

- EIB síť s jednou linií; tato EIB síť obsahuje funkční prvky ABB a uživatelské prvky Gira.
- Průmyslový počítač **DataLab PC** s napájením 24V DC.
- Rozhraní **DataLab IF/EIB** připojení k počítači **DataLab PC** USB kabelem.
- Průmyslový řídicí a měřicí systém **Control Web** s aplikací řešící všechny rozšiřující požadavky řízení pomocí EIB.



Jaká je použita v tomto bytě základní filozofie provozu? Velmi jednoduchá. Byt může přecházet mezi stavy zamknut a odemknut (a samozřejmě, o přechodu mezi těmito stavy musí rozhodovat člověk) a ostatní funkce bytu jsou tomuto

hlavnímu stavu přizpůsobeny. Některé skutečnosti shrnuje další přehled:

- **vytápění** je omezeno na udržování základní teploty (18°C),



- **osvětlení** večer simuluje přítomnost obyvatel (používají se různé uchované vzorky provozu světel s náhodně posouvány okamžiky rozsvěcování a zhasínání),

- pohybová čidla, čidla přítomnosti a okenní spínače sledují **zabezpečení** a v případě porušení vyvolávají poplach.

Vše výše uvedené (kromě simulace přítomnosti pomocí světel) je realizováno pouze s pomocí EIB. Zatím není použito nic, co by vyžadovalo přítomnost počítače spolu s programovou aplikací systému **Control Web**. Je to jasné — všechno výše popsané je v prostředí EIB přirozené a jednoduché.

K čemu pak je Control Web a počítač DataLab PC využít?

K internetové komunikaci s okolím

Součástí aplikace systému **Control Web** je přístroj *httpd* (internetový server), pomocí něhož lze do vybraných částí aplikace nahlížet a stejně vybrané části aplikace i ovládat. Přístup k tomuto serveru je samozřejmě zabezpečen, rovněž prostředky systému **Control Web** — aplikace sleduje, odkud se klient pokouší k serveru připojit a aktivní spojení uzavře pouze



s klientem, kterého považuje za důvěryhodného.

K informování o neočekávaných stavech

V tomto případě pracuje aplikace systému jako http klient (klient www serveru). V případě, kdy dojde v bytě k něčemu podivnému, použije aplikace http klienta k odeslání textové zprávy do SMS serveru, která je pak dále přeposlána příslušnému mobilnímu telefonu (to již ale zřejmě neřeší aplikace v bytě).

Typicky se za neočekávaný stav považuje porušení zabezpečení bytu, takže dojde-li například k pokusu byt vyloupit, systém ihned (pomocí SMS) informuje.

K simulaci přítomnosti osob pomocí světel

O této vlastnosti aplikace se již výše píše — aplikace v sobě tu a tam uchová sekvenci rozsvěcování a zhasínání světel, načtež v případě, že je byt zabezpečen, tuto sekvenci večer samostatně přehrává. Přestože jsou takového chování schopny i speciální přístroje EIB, je v tomto případě efektivnější dosahovat stejné funkce programovými prostředky.

Ke korekcím nelinearit termostatických ventilů V bytě jsou jednotlivá topná tělesa (ventily) proporčně ovládány. V jednotlivých místnostech je více topných těles, a je proto nutné, aby tato tělesa topila stejně. Praxe ukázala, že termostatické ventily, určené k použití termostatických hlavice, jsou velmi nevyrovnané a jeden od druhého se velmi liší. Výrobci evidentně počítají

s tím, že termohlavice (měřící teplotu ve svém nitru) jakkoli nepřesným ventilem nakonec správný průtok propustí. Bez termohlavice je však nutné termostatické ventily korigovat jinak, což se v tomto případě děje programově.

K uchování historie teplot

Tato část aplikace není nutná z funkčních důvodů, jedná se zde o přírodovědný zájem. Aplikace sleduje venkovní teplotu (pomocí EIB přístrojů) každých 5 minut, načtež tyto údaje ukládá (a pak samozřejmě nabízí k prohlížení) a dále z nich počítá denní teplotu, kterou také následně archivuje.



Na závěr

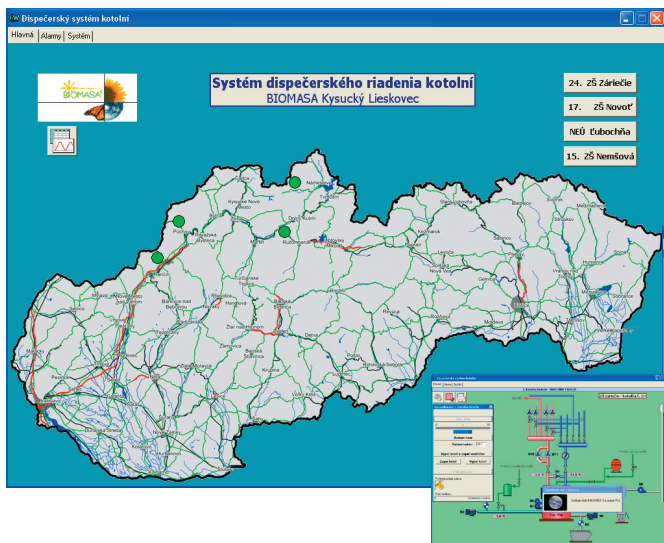
Popsané spojení EIB se systémem **Control Web** a počítačem **DataLab PC** je ukázkou malého, efektivního nasazení moderních systémů a jejich vzájemného propojení. Je zjevné, že použití programu přidává k již docela mocné EIB síti další sofistikovanou funkčnost — v popsáném konkrétním případě integrálně do bytu zapojenou. Použití řešení na jedné straně redukuje zaběhnuté principy a na straně druhé dodává jednomu řešení mnohočetný smysl — pěkně to ukazuje příklad zabezpečení bytu: zaběhnuté řešení by použilo samostatná vedení, čidla a ústřednu, použité řešení využívá jedněch a těchž přístrojů jednou pro zabezpečení a jednou pro běžné funkce. Změna chování je přítom jen informační (programová) — a takové změny jsou přirozeně velmi snadné.

Martin Kolařík, kolarik@mii.cz

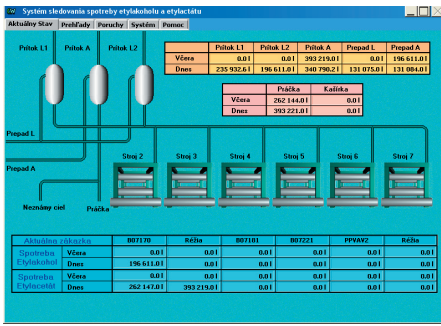
Několik ukávek aplikačních programů

Dodavatelem uvedených systémů je SOFOS-X, Jesenského 16, 010 01 Žilina

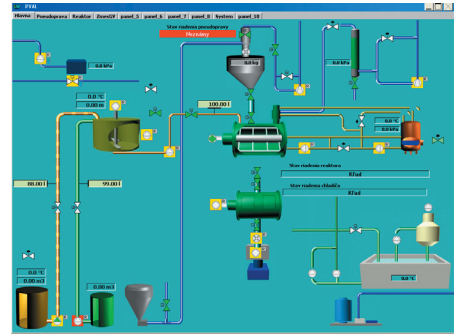
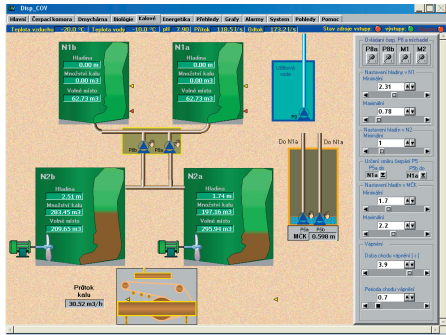
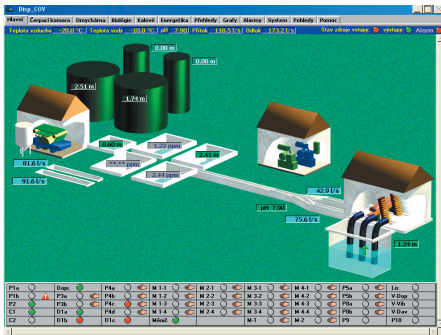
Uvedme si zde ukázky několika aplikací systému **Control Web**, které se vyznačují hezkým grafickým řešením a velmi kvalitním uživatelským rozhraním. Komunikují s různými PLC i se zásuvnými kartami, pracují v různém síťovém i databázovém prostředí a mohou být dobrou inspirací pro naše další řešení.



Zákazník	BIOMASA, a.s.
DB	Access
PLC	DataLON
kanály	cca 50 na jednu kotelnou
Funkčnost	Uživatel provozuje kotelnou po celém Slovensku. Systém je vybaven modemem a na požádání obsluhy se spojí s příslušnou kotelnou a načte aktuální data včetně poruchových stavů. V případě poruchy se kotelnou automaticky připojí a obsluhu se zobrazí příslušný alarmový stav.

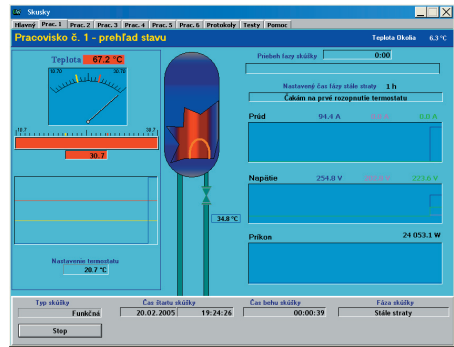
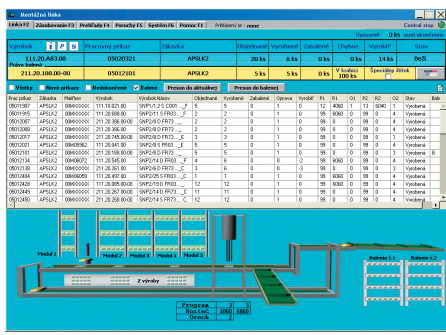
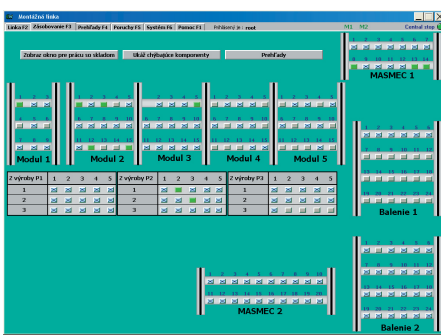


Zákazník	Chemosvit, a.s. SVIT
DB	MSDE
PLC	SC-552C
Komunikace	SC BUS
Kanály	cca 150
Funkčnost	<ul style="list-style-type: none"> o měření spotřeby ředidel na jednotlivé tiskárenské stroje o připojena stanice na hlášení alarmových koncentrací výbušných plynů o měření průtoků o přehledy o spotřebě ředidel na jednotlivé stroje, zakázky a období o poskytování přehledů do podnikové sítě v roli http serveru

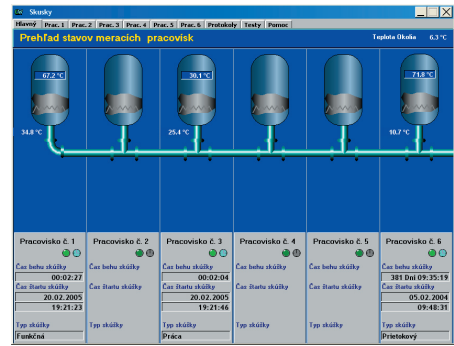


Zákazník	VHOS, a.s. a Město Moravská Třebová
DB	MSDE
PLC	Alfa Laval SattCon35 – 2 uzly
Komunikace	SattBus
Kanály	cca 800
Funkčnost	<ul style="list-style-type: none"> o komplexní řízení čistírny odpadních vod o sledování spotřeby elektrické energie o přehledy o protečených množstvích a spotřebě elektrické energie za libovolné období o v noci a přes víkendy bezobslužný provoz o v době nepřítomnosti obsluhy hlášení kritických poruch pomocí SMS

Zákazník	PANARA, s.r.o. NITRA
DB	Access
PLC	Alfa Laval SattCon 200, SattScale
Komunikace	SC BUS
Kanály	cca 1500
Funkčnost	řízení výroby AL folie



Zákazník	Sauer-Danfoss, a.s. Považská Bystrica
DB	Access
PLC	SIMATIC S7-300
Další zařízení	tiskárna štítků ZEBRA, čtečka čárového kódu, značkovač TELESIS, OP 17, šroubovací stanice BOSCH, druhý monitor
Komunikace	MPI
Kanály	cca 450
Funkčnost	<p>Dvoumonitorový systém řídí montážní linku čerpadel. Na jednom monitoru je aplikace se zadáváním příkazů a obsluhou skladu. Druhý monitor je přímo uprostřed linky a zobrazuje informace o aktuální výrobě s poskytováním technické dokumentace jako jsou technické výkresy, speciální instrukce pro obsluhu a pracovní postupy.</p> <ul style="list-style-type: none"> o skladové hospodářství <ul style="list-style-type: none"> ▪ na základě databáze výrobků z podnikového informačního systému sleduje potřebu komponentů ▪ řídí LED regálového skladu o určuje a nastavuje programy pro šroubovací stanice o ovládá pracovní příkazy se čtečkou čárových kódů o ovládá balicí pracoviště přes OPI7 a tisk balicích štítků o ovládá značkovač tělesa čerpadla o poskytuje přehledy o výrobě a balení do podnikové sítě jako http server



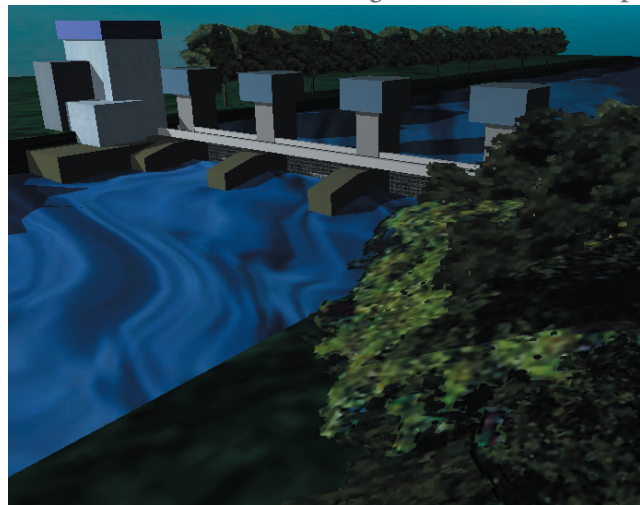
Zákazník	TATRAMAT, a.s. Poprad
DB	Access
PLC	ADVANTECH karty a SCM
Komunikace	ADVANTECH PLC, SC BUS
Kanály	cca 80
Funkčnost	<ul style="list-style-type: none"> o řízení zkoušek ohřivačů vody o tvorba a tisk protokolů u zkouškách

Krok vpřed při vizualizaci procesů a operátorském řízení

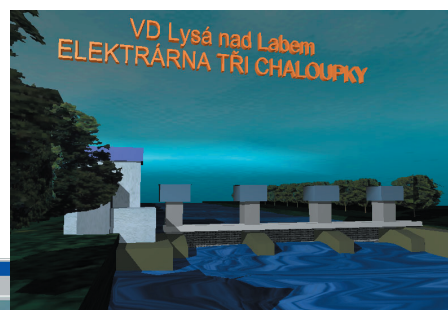
SCADA aplikace se již mnoho let navzájem podobají jako vejce vejci. Obvykle obsahují nějaká technologická schémata a k nim pár grafů a tabulek. V řadě případů může být právě taková strohá vizualizace optimálním řešením, neméně často však 3D grafika

může aplikaci přidat na přehlednosti a vizuální atraktivitě. Využitím moderních technologií počítačové grafiky se aplikační programy posouvají na vyšší technickou úroveň. To, že veškeré SCADA programy přečtou z řídicích PLC data a správně je zobrazí na moni-

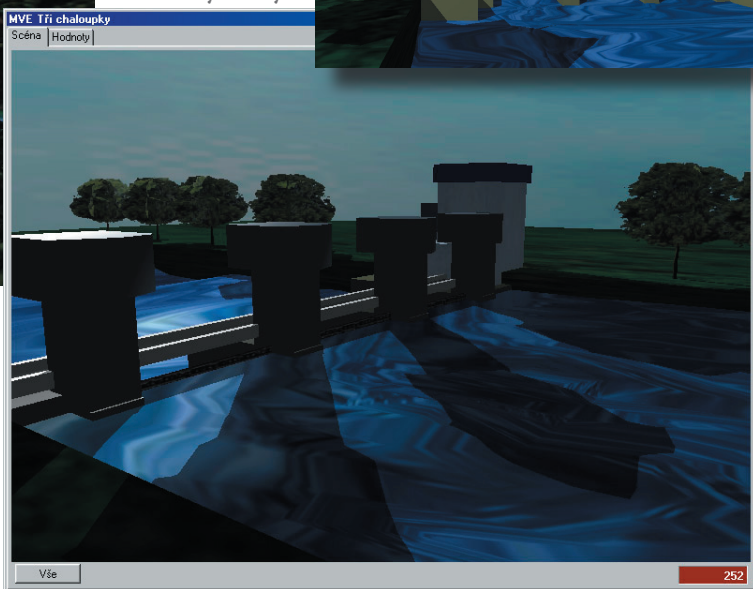
nemohou nijak lišit. Rychlá a realistická třírozměrná vizualizace je ale v dnešní průmyslové automatizaci stále něčím navíc, je proti většině současných systémů pro operátorské řízení významnou přidanou hodnotou.



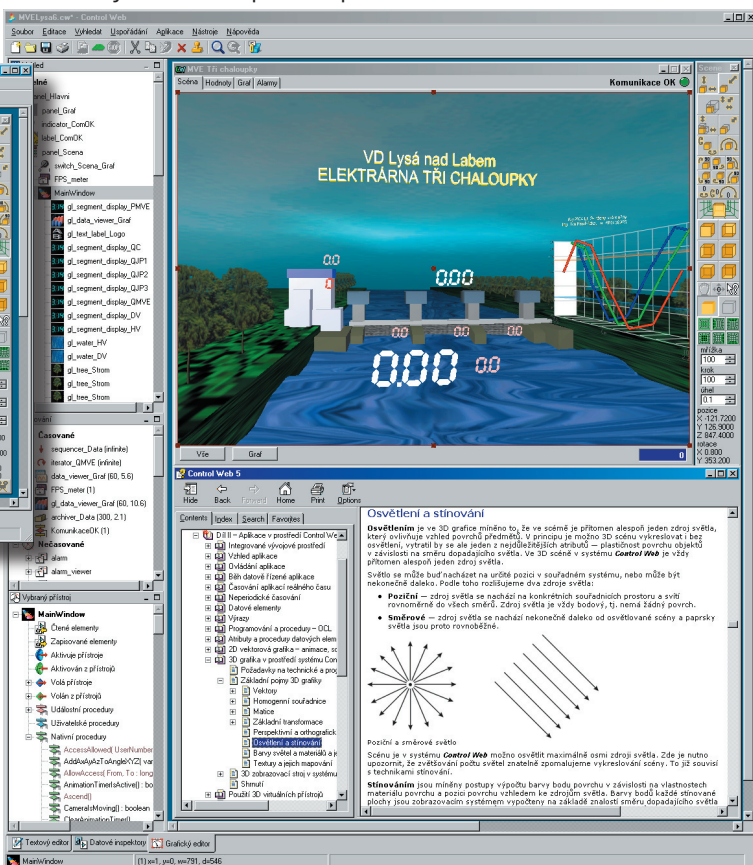
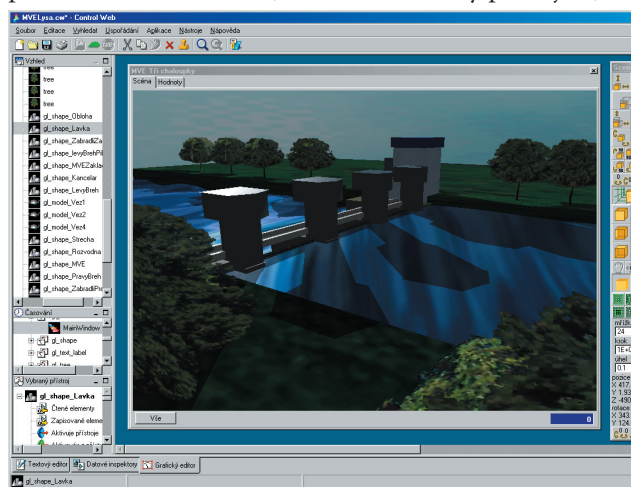
toru a případně uloží do patřičných databází je naprostou samozřejmostí - v tom se jednotlivé systémy



Uživatelské rozhraní řídicího systému vodního díla v Lysé nad Labem je koncipováno klasicky jako okno s překrývajícími se panely přepínanými řadou vodorovných záložek. Pouze jeden panel, a to titulní, je třírozměrnou scénou s perspektivní projekcí. Ke zvýšení efektu vizualizace přispívá vlnící se vodní hladina se sféricky mapovanou odleskovou texturou. Scéna není náročná na výkon počítače, i se základními levnými grafickými kartami je dosahováno několika desítek FPS a s lepšími kartami mnoha set FPS. A to dokonce i se zapnutými prostorovými stíny a animovaným kymácením stromů ve větru pomocí vertex shaderů (o tom ale až někdy později ...).



Vizualizace jezu za běhu aplikace a při editaci v IDE CW5.



Dodavatelem programového vybavení popisovaného na této dvoustraně je:

Ing. Jan Procházka
ELIS - inženýrské služby
Šafaříkova 666
500 02 Hradec Králové

Telefon: 495 513 055, 495 531 557
Fax: 495 513 054, 495 531 558
e-mail: info@elis-hk.cz

Programové vybavení pro řídicí systém plavební komory v Českých Kopistech výborně demonstruje elegantní a bezproblémové začlenění třírozměrné vizualizace do rozsáhlého klasického dvourozměrného grafického uživatelského rozhraní, sestávajícího z obvyklých tabulek alarmů a grafů průběhů veličin i dvourozměrné technologické schéma. Prostorová scéna zde zabírá celkem malou část obrazovky. Pro lepší orientaci operátora má scéna definovány tři přednastavené pozice kamery. S poměrně jednoduchými prostředky je zde dosaženo velmi dobrého efektu.

Řídicí systém VPK České Kopisty
10. 12. 2004 13:16:57

Stav	Popis	Datum-vznik	Čas-vznik	Datum-zánik	Čas-zánik	Operátor-povolení
NEAKT	Blokace automatu	10.12.2004	12:03:50,791	10.12.2004	12:04:46,796	
NEAKT	Vypadek jističe RH1.1.2	10.12.2004	08:25:57,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe semaforu VPK	10.12.2004	08:25:56,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe RM2	10.12.2004	07:30:16,781			
NEAKT-P	Vypadek jističe semaforu VPK	10.12.2004	07:30:14,781			
NEAKT	Vypadek jističe přívodu RH1.1	10.12.2004	07:30:14,781			

VELKÁ PLOVEBNÍ KOMORA
OVLÁDÁNÍ Z MÍSTA

Nahoru: NAHORU
Dolů: DOLU
Plavební stop: PLAVEBNÍ STOP
Porucha: STOP

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

HORNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ OHLAVÍ: 143.84
146.58
146.60
HORNÍ OHLAVÍ

Dynamická ochrana: ZARAZENA, VYRAZENA
Dolní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Dolní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Rychlost: 0.0

Řídicí systém VPK České Kopisty
10. 12. 2004 13:21:13

Stav	Popis	Datum-vznik	Čas-vznik	Datum-zánik	Čas-zánik	Operátor-povolení
NEAKT	Blokace automatu	10.12.2004	12:03:50,791	10.12.2004	12:04:46,796	
NEAKT	Vypadek jističe RH1.1.2	10.12.2004	08:25:57,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe semaforu VPK	10.12.2004	08:25:56,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe RM2	10.12.2004	07:30:16,781			
NEAKT-P	Vypadek jističe semaforu VPK	10.12.2004	07:30:14,781			
NEAKT	Vypadek jističe přívodu RH1.1	10.12.2004	07:30:14,781			

Řídicí systém VPK České Kopisty
10. 12. 2004 13:24:09

Stav	Popis	Datum-vznik	Čas-vznik	Datum-zánik	Čas-zánik	Operátor-povolení
NEAKT	Blokace automatu	10.12.2004	12:03:50,791	10.12.2004	12:04:46,796	
NEAKT	Vypadek jističe RH1.1.2	10.12.2004	08:25:57,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe semaforu VPK	10.12.2004	08:25:56,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe RM2	10.12.2004	07:30:16,781			
NEAKT-P	Vypadek jističe semaforu VPK	10.12.2004	07:30:14,781			
NEAKT	Vypadek jističe přívodu RH1.1	10.12.2004	07:30:14,781			

VELKÁ PLOVEBNÍ KOMORA
OVLÁDÁNÍ Z VELINU

Nahoru: NAHORU
Dolů: DOLU
Plavební stop: PLAVEBNÍ STOP
Porucha: STOP

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

HORNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ OHLAVÍ: 143.83
146.62
146.62
HORNÍ OHLAVÍ

Dynamická ochrana: ZARAZENA, VYRAZENA
Dolní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Dolní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Rychlost: 0.0

Řídicí systém VPK České Kopisty
10. 12. 2004 13:23:25

Stav	Popis	Datum-vznik	Čas-vznik	Datum-zánik	Čas-zánik	Operátor-povolení
NEAKT	Blokace automatu	10.12.2004	12:03:50,791	10.12.2004	12:04:46,796	
NEAKT	Vypadek jističe RH1.1.2	10.12.2004	08:25:57,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe semaforu VPK	10.12.2004	08:25:56,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe RM2	10.12.2004	07:30:16,781			
NEAKT-P	Vypadek jističe semaforu VPK	10.12.2004	07:30:14,781			
NEAKT	Vypadek jističe přívodu RH1.1	10.12.2004	07:30:14,781			

Měření elektrických veličin

Uz	U [V]	I [A]	P [kW]	Q [kVAR]	S [kVA]	Cos FI
U1N	238					
U2N	237					
U3N	238					
U12	410					
U23	410					
U31	407					
I1	539					
I2	465					
I3	448					
P činný	36					
Q jalový	11					
S zdánlivý	38					
Cos FI	100					

Řídicí systém VPK České Kopisty
10. 12. 2004 13:22:18

Stav	Popis	Datum-vznik	Čas-vznik	Datum-zánik	Čas-zánik	Operátor-povolení
NEAKT	Blokace automatu	10.12.2004	12:03:50,791	10.12.2004	12:04:46,796	
NEAKT	Vypadek jističe RH1.1.2	10.12.2004	08:25:57,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe semaforu VPK	10.12.2004	08:25:56,781	10.12.2004	08:26:26,781	
NEAKT	Vypadek jističe RM2	10.12.2004	07:30:16,781			
NEAKT-P	Vypadek jističe semaforu VPK	10.12.2004	07:30:14,781			
NEAKT	Vypadek jističe přívodu RH1.1	10.12.2004	07:30:14,781			

KALIBRACE MĚŘENÍ HLADIN

Uz	U [V]	I [A]	P [kW]	Q [kVAR]	S [kVA]	Cos FI
U1N	238					
U2N	237					
U3N	238					
U12	410					
U23	410					
U31	407					
I1	539					
I2	465					
I3	448					
P činný	36					
Q jalový	11					
S zdánlivý	38					
Cos FI	100					

VELKÁ PLOVEBNÍ KOMORA
OVLÁDÁNÍ Z VELINU

Nahoru: NAHORU
Dolů: DOLU
Plavební stop: PLAVEBNÍ STOP
Porucha: STOP

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

HORNÍ SEMAFORY OVLÁDÁNÍ: ZÁKAZ, PRŘR, POVOL, Odstav, ZÁKAZ, POVOL

SEMAFORY: VJEZDOVÝ, VYJEZDOVÝ

DOLNÍ OHLAVÍ: 143.78
148.57
HORNÍ OHLAVÍ

Dynamická ochrana: ZARAZENA, VYRAZENA
Dolní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Dolní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní odtoky: OTEVŘÍ, ZAVŘÍ, STOP
Horní vrata: OTEVŘÍ, ZAVŘÍ, STOP
Rychlost: 0.0



Pro vizualizaci technologie malé vodní elektrárny Padrtý bylo využito pouze hotových komponent dodávaných v rámci systému **Control Web 5**. Pro velkou řadu vizualizací v oblasti průmyslové automatizace není nutno pomocí 3D modelovacích programů vytvářet vlastní modely. Tvorbou modelů bezesporu patří k docela složitým a tudíž také k celkem neoblíbeným činnostem. Už jen pochopení základních principů ovládaní takových programů jako je např. 3DS Max, Cinema 4D, Blender atd. zabere několik dní času. Prostředí **Control Web 5** je vybaveno množstvím 3D komponent i s předpřipravenými animacemi. Řadu vizualizací tak lze sestavit velmi rychle a efektivně.

Kód	Produkt	Cena pro integrátory	Koncová cena
-----	---------	----------------------	--------------

Control Web 5

CW5-DEV	Control Web 5 Vývojová verze	19 700 Kč	21 700 Kč
CW5-UCW4	Control Web 5 Vývojová verze upgrade z Control Web 2000	12 900 Kč	14 200 Kč
CW5-SRUN	Control Web 5 Runtime	5 900 Kč	6 500 Kč
CW5-NRUN	Control Web 5 Runtime Network Edition pro provozování síťových distribuovaných aplikací	9 700 Kč	10 700 Kč
CW5-TWCW	Control Web 5 Technologická záruka bude poskytnuta po předložení poukazu		500 Kč
CW5-DEMO	Control Web 5 Demonstrační verze na CD-ROM lze zdarma stáhnout z http://www.mii.cz		250 Kč

Ovladače dodávané spolu se systémem

CW5-NET	Ovladač pro komunikaci v síti přes protokol TCP/IP	zdarma	zdarma
CW5-DLUSB	Ovladač pro komunikaci s USB moduly DataLab I/O vyžaduje Windows 2000 a vyšší	zdarma	zdarma
CW5-ASCII	Univerzální ovladač pro textovou komunikaci přes sériové rozhraní RS-232	zdarma	zdarma
CW5-DDECL	Univerzální DDE klient	zdarma	zdarma
CW5-DLEIB	Ovladač rozhraní DataLab IF/EIB vyžaduje Windows 2000 a vyšší	zdarma	zdarma

DataLab PC

DL-PC500	DataLab PC 500 256 MB PC133 SDRAM	10 700 Kč	11 750 Kč
DL-PC600	DataLab PC 600 256 MB DDR266 SDRAM	11 900 Kč	13 100 Kč
DL-PC610	DataLab PC 610 256 MB DDR266 SDRAM	13 400 Kč	14 750 Kč
DL-LM15T	DataLab LCD 15T 15" LCD monitor s dotykovou obrazovkou	27 900 Kč	30 690 Kč
DL-LM15T	DataLab LCD 15T 15" LCD monitor	22 700 Kč	25 000 Kč

DataLab PC/IO

DL-IO500	DataLab PC/IO 500 256 MB PC133 SDRAM + CPU modul jednotky DataLab IO	12 970 Kč	14 250 Kč
DL-IO600	DataLab PC/IO 600 256 MB DDR266 SDRAM + CPU modul jednotky DataLab IO	14 170 Kč	15 600 Kč
DL-IO610	DataLab PC/IO 610 256 MB DDR266 SDRAM + CPU modul jednotky DataLab IO	15 670 Kč	17 250 Kč

DataLab IO

DL-CPU	DataLab IO skříňka + CPU	2 970 Kč	3 250 Kč
DL-CPUM	DataLab IO μ skříňka + CPU	1 930 Kč	2 100 Kč
DL-DI1	DataLab modul DI8 8 digitálních izolovaných vstupů	1 350 Kč	1 500 Kč
DL-DI2	DataLab modul DI8SE 8 digitálních izolovaných vstupů, společná zem	1 350 Kč	1 500 Kč
DL-DO2	DataLab modul DO8 8 digitálních izolovaných výstupů	1 350 Kč	1 500 Kč
DL-DO1	DataLab modul RO8 8 releových výstupů, spínací	1 500 Kč	1 650 Kč
DL-AI1	DataLab modul AI8 8 analogových vstupů, 16 bitů	2 690 Kč	2 950 Kč
DL-AO1	DataLab modul AO8 8 analogových výstupů, 12 bitů	2 890 Kč	3 200 Kč
DL-EIB	DataLab EIB EIB/USB	3 700 Kč	4 050 Kč

Zde je uvedeno jen několik položek z rozsáhlého ceníku. Kompletní nabídku produktů naleznete na <http://www.mii.cz>. Výhodné a velmi pohodlné je využívání služeb internetového obchodu. Můžete nejen evidovat vaše objednávky, ale i sledovat jejich stav včetně kompletní historie nákupů. Internetový obchod také můžete využít pro kalkulaci cen zvolených sestav, stačí si jen vytisknout neodeslanou objednávku.