

PROGRAMOVÝ SYSTÉM CONTROL WEB A JEHO MOŽNOSTI INTERAKCE S REÁLNÝM PROSTŘEDÍM

Roman Cagaš, Pavel Cagaš, Jiří Kofránek

Anotace

Control Web je distribuované objektové prostředí pro vývoj aplikací reálného času, vizualizaci procesů a simulací s využitím platformy Windows, vyvíjená více než 20 let. V průmyslových aplikacích se Control Web stal v ČR de facto velmi rozšířeným standardem. Pro tvorbu biomedicínských přístrojů nabízí možnosti snadného vizuálního návrhu jednotlivých řídicích a vizualizačních komponent biotechnologických přístrojů. Jednou z dalších možností je i využití systému pro vizualizaci výukových simulací. Předchozí verzi tohoto systému byla např. využita pro realizaci simulátoru Golem nebo a tvorbu interaktivního simulátoru ledviny. Současná verze systému mimo jiné umožňuje využívat 3D komponenty a nástroje počítačového vidění. Pro tuto verzi je vyvíjena simulační komponenta, která umožní využít vizualizaci výstupů simulačního modelu běžícího na pozadí systému. Práce na tvorbě výukových lékařských simulátorů je mimo jiné podporována projektem MPO FR—TI3/869.

Klíčová slova:

Control Web, průmyslová automatizace, simulace, strojové vidění, řízení, vizualizace

1. Úvod

V roce 1990, v době velkých nadějí a očekávání, kdy se otevřely možnosti podnikání, skupina mladých „valašských ogařů“ založila akciovou společnost, kde chtěla realizovat své sny o vytvoření vlastního technologického podniku, kde budou společným úsilím vyvíjet nové, špičkové a na trhu úspěšně prodávané výrobky.

Od té doby „ogaři“ zestárlí, někdejší velká porevoluční euforie se již dávno rozplynula, ale malá, inovativní a průbojná společnost Moravské přístroje již více dvacet let vytváří hi-tech výrobky, které jsou po technologické stránce plně srovnatelné se světovou špičkou. K těmto výrobkům patří vývojové prostředí Control Web, které je v současné podobě výsledkem více než dvacetiletého vývojového úsilí na poli nástrojů pro průmyslovou automatizaci.

2. Současné požadavky na softwarové nástroje pro průmyslovou automatizaci

Ještě v nedávné minulosti byl svět programového vybavení pro průmyslovou automatizaci jednodušší. Existovaly jasně rozdělené oblasti působnosti jednotlivých kategorií programového vybavení:

- Na nejnižší úrovni to byly programy pro jednočipové řadiče nebo vestavěné řídicí počítače (firmware). Tyto programy mívají většinou výrobcem zařízení již pevně danou funkčnost, případně bývají jen velmi omezeně parametrizovatelné.

- Prostředním článkem bývaly obecně použitelné programovatelné automaty (PLC – Programmable Logic Controller) nebo specializované numerické řídicí jednotky strojů (CNC – Computer Numerical Control). Programovací model PLC je obvykle velice prostý – časovací jádro periodicky spouští sekvence interpretovaných instrukcí. Programovací jazyky do jisté míry připomínají instrukce programovatelných kalkulaček.
- Na vrcholu stávaly systémy pro operátorské řízení technologického procesu (SCADA – Supervisory Control and Data Acquisition). Principy fungování těchto tzv. vizualizačních programů se většinou nezměnily – základem je tabulka datových elementů (či tagů, datových bodů, atd., každý výrobce má své pojmenování), jejíž data jsou cyklicky obnovována buď čtením hodnot z periferních zařízení nebo výpočtem z jiných elementů. Možnosti programování bývají velmi omezené, vesměs se jedná o pouhou parametrizaci fixního programu.

Toto členění a z něj plynoucí principy výstavby automatizačních projektů přetrvávaly dlouhá desetiletí. Každý člověk z oboru se v této struktuře dobře vyznal, věděl, co může od každé kategorie zařízení čekat, a také obvykle nevznikaly žádné pochyby, jak nový projekt koncipovat.

Doba ale pokročila, používáme mobilní telefony, přenosné počítače, kdekoliv právě jsme, máme k dispozici bezdrátové připojení na Internet, a především – všechna zařízení navzájem komunikují. Zde nám velký užitek přinášejí standardní a všeobecně přijaté komunikační protokoly a standardní programová rozhraní. Je docela pochopitelné, že uživatelé vyžadují takový komfort a vlastnosti, na které jsou zvyklí i po řešeních v oblasti průmyslové automatizace. Systémy vytvořené klasickými „historickými“ prostředky mají pro dnešní dobu informačních a komunikačních technologií několik zatěžujících vlastností:

- Většina zařízení komunikuje prostřednictvím firemních proprietárních, často až nepochopitelně tajných a právně chráněných protokolů. Uživatel se tak stává „rukojmím“ dodavatele zařízení a nemůže systém udržovat a rozvíjet podle svých představ, které se s časem a rozvojem technologií přirozeně mění
- Používané programové vybavení není škálovatelné – pro různé kategorie aplikací je nutno používat různá vývojová prostředí a různé programovací modely a jazyky.
- Programové vybavení je obvykle svázáno se zařízeními stejného výrobce, není snadno propojitelné se systémy jiných dodavatelů a je obtížně zařaditelné pro informačního systému podniku.
- Systémy jsou jen obtížně rozšiřitelné o nová zařízení jiných výrobců.
- Funkčnost běžná v informačních technologiích (např. webová rozhraní, SMS zprávy, GPRS připojení, bezdrátové operátorské panely atd.) je realizovatelná jen komplikovanými způsoby, kter0 celý systém citelně prodražují. Tyto technologie obvykle nejsou přirozenou součástí použitých prostředků.

- Celé systémy se obvykle skládají z většího počtu zařízení a jsou podstatně dražší. Rovněž vývoj a další údržba programového vybavení jsou velmi nákladné.

3. Control Web a jeho předchůdci

Počátkem devadesátých let se díky masové výrobě dnes „pécéčko“ stala jednou z nejlevnějších periférií technologického zařízení. Proto se na trhu objevila řada výše zmíněných SCADA systémů, které sloužily k rychlému a flexibilnímu vytvoření vizualizačních měřících nebo řídicích aplikací. Tímto způsobem bylo možné poměrně rychle a levně vytvořit „velín“ z obyčejného PC, vybaveného příslušným adaptérem (I/O kartou), komunikující s technologickým zařízením.

Valná část těchto systémů pracuje na jednoduchém principu: vytvořený program pracuje cyklicky v určité smyčce, kde postupně obhospodařuje sadu záznamů, v nichž jsou uschovávána veškerá (vstupní či výstupní) technologická data. Při aktivaci těchto záznamů mohou být technologická data v číselné či grafické podobě vizualizována na obrazovce PC, či je naopak iniciován přenos řídicích dat do připojené technologické aparatury. Systémy poskytují různé možnosti interaktivního vytváření grafické podoby obrazovek „velínů“ z jednotlivých virtuálních přístrojů (měřících ručičkových či digitálních displejů, obrazovek pro průběžné zobrazování grafů, tlačítek, přepínačů apod.). Podstata „programování“ pak spočívá v zařazování aktivace těchto přístrojů do nekonečné smyčky řídicího programu.

Problém je ale v tom, že obvykle není možné příliš ovlivnit rychlost, strukturu a mechanismus čtení a záznamů dat do vstupních a výstupních zařízení, což při menších objemech veličin příliš nevádí. Jakmile je však počet načítaných či odesílaných dat větší, dostává se systém do problémů – některé elementy jsou obsluhovány příliš často a na mnoho jiných se zase včas nedostává. Má-li systém pracovat v reálném čase, neznamená to, že stihne úplně všechno – program je snadno možno přetížit a pak se stane, že vnitřní smyčka nestihne včas doběhnout do potřebného začátku dalšího spuštění. Autor aplikace však musí mít časování systému zcela pod kontrolou a musí mít diagnostiku, která umožní vše monitorovat a aplikaci vyladit tak, aby se nedostávala do časového skluzu.

Proto vývojový tým Moravských přístrojů začal pracovat na programovém systému, který nebude omezen jen na vizualizaci a sběr dat na operátorských pracovištích, ale umožní vytvářet aplikace pro přímé řízení strojů v reálném čase, komplexní aplikace v oblasti simulace a modelování, zpracování dat v počítačových sítích atd.

3.1 Control Panel – programové komponenty ve výkoném víceúlohovém systému

V době vzniku tohoto produktu zde byl operační systém Windows 3.1 jako nadstavba nad DOSem, a tento operační systém jistě nebyl tím pravým prostředím pro trvalý provoz řídicích aplikací v reálném čase v průmyslovém prostředí. Proto byl pro **Control Panel** vytvořen vlastní operační systém,

který pracoval v chráněném módu procesorů 286 a vyšších, virtualizoval paměť kódu i dat (pro data vytvářel odkládací soubor) a mohl spustit mnoho paralelně běžících úloh. Tento systém byl vybaven velmi kvalitním grafickým uživatelským rozhraním pracujícím s libovolně složitou hierarchickou strukturou oken na obrazovce¹.

Systém byl i na tehdejší hardware velice výkonný a především stabilní, aplikace v prostředí Control Panel bývaly běžně provozovány v nepřetržitých prozozech několik let bez odstávky.

Také struktura systému Control Panel a principy tvorby aplikačního programu byly přelomové.

Místo neovlivnitelného (pouze parametrizovatelného) vnitřního cyklu bylo možné vytvářet volně programovatelnou aplikaci, kde se dala programátorovi do rukou možnost určit, kdy se má ta či ona hodnota načítat či zapisovat do vstupně-výstupního zařízení, a to nejen pomocí nějakých pevných časových konstant, ale např. i jako reakce na některé události.

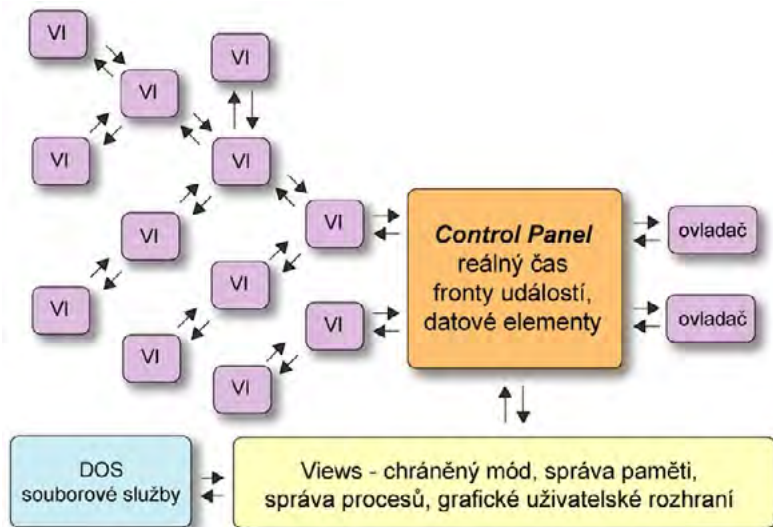
V systému existovaly volně definovatelné datové elementy a vizualizační prvky – tzv. virtuální přístroje, které dokázaly s těmito elementy pracovat. Virtuální přístroje zde byly externími programovými komponentami, jejichž počet a vlastnosti nejsou systému při budování struktury aplikačního programu známy. Struktura aplikačního programu v Control Panelu byla tedy tvořena obecným stromem instancí virtuálních přístrojů, které jsou navzájem propojeny do stromů viditelnosti na obrazovce a do struktury časování a toku událostí. Pro každý virtuální přístroj byla přístupná kompletní množina instancí datových elementů. Virtuální přístroje neví nic o tom, jak tyto datové elementy získávají data z periferních zařízení a pracují abstraktně se všemi elementy shodně.

S vnějším světem systém komunikoval pomocí ovladačů (programových komponent s abstraktním, i když v tomto případě celkem jednoduchým, procedurálním rozhraním) prostřednictvím speciálních proměnných – tzv. „vstupních a výstupních kanálů“, které se vždy nadefinovaly pro daný ovladač. Prostřednictvím těchto kanálů se (přes příslušný ovladač) přenášely hodnoty mezi periferním zařízením a instancemi vizualizačních komponent (virtuálních přístrojů).

Při vývoji aplikace byla definována a popsána struktura instancí všech typů komponent včetně všech jejich vzájemných vazeb. Systém Control Panel tehdy pouze dokázal z aktuálně přítomných knihoven vytvářet a propojovat instance programových komponent do podoby aplikačního programu.

V dalších verzích systému bylo virtuálním přístrojům přidáno i dynamicky, tj. za běhu aplikace, detekovatelné rozhraní a vznikla tak další úroveň komunikace mezi jednotlivými programovými komponentami. Bylo tak možno vytvářet

1. V té době byl dokonce vytvořen a prodáván kancelářský systém obsahující textový procesor, tabulkový procesor a kreslicí nástroj jako nadstavba nad DOSem – z mnoha důvodů samozřejmě nemohl obstát v konkurenci s Microsoftem, ale pravdou je, že „okna ze Zlína“ byly v té době stabilnější než Windows ze Seattlu.



Obrázek 1 — Struktura aplikace v prostředí Control Panel

algoritmy i způsobem připomínajícím tradiční imperativní programovací jazyky, i když v tomto případě sekvence příkazů probíhaly na vysoké úrovni nad strukturou instancí mnohdy velmi komplexních komponent. Vznikl tak základ pro pružný a všestranný programovací jazyk OCL (Object Control Language).

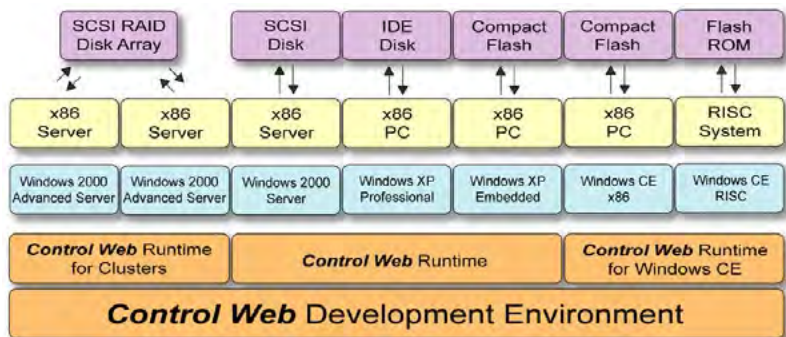
Control Panel také zavedl technologii vizuálního programování, kdy program v Control Panelu bylo možné vytvářet v grafickém prostředí, pomocí myši vybírat jednotlivé komponenty a rozmisťovat je na zobrazitelnou plochu či zařazením do stromů časování definovat jejich aktivaci v programu z hlediska času a v inspektorech jednotlivých komponent interaktivně pomocí dialogů definovat příslušné vlastnosti apod.

Zároveň bylo možné celý systém překlopit z grafické do textové podoby a pokračovat v tvorbě programu v textovém režimu. Z textového režimu ale bylo vždy možné se kdykoli překlopit (po opravě všech případných syntaktických chyb) zpět do grafického režimu. Každou aplikaci je proto možné tvořit chvíli v textovém a chvíli v grafickém režimu, vždy podle potřeby.

Možnosti této technologie byly s vlastnostmi tehdy používaných SCADA programů nesrovnatelně mocnější a Control Panel byl uživateli velmi pozitivně přijat a sloužil všude tam, kde schopnosti tradičních systémů již nedosahovaly.

3.2 Příchod 32-bitových Windows a systém Control Web

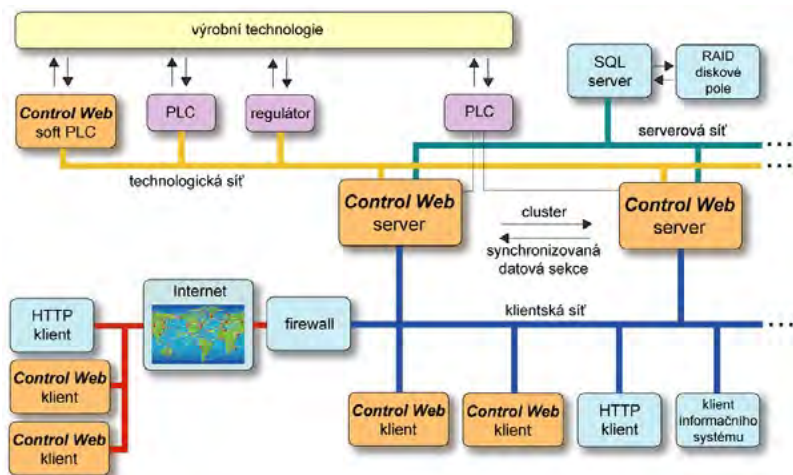
S příchodem Windows NT se situace změnila. Byl zde stabilní operační systém, který byl dobře použitelný i pro trvalý běh aplikací v náročném nepřetržitém průmyslovém provozu. Existující systém **Control Panel 3.1** byl převeden pod Windows pod názvem **Control Web 3.1**. Začala tak nová etapa rozvoje



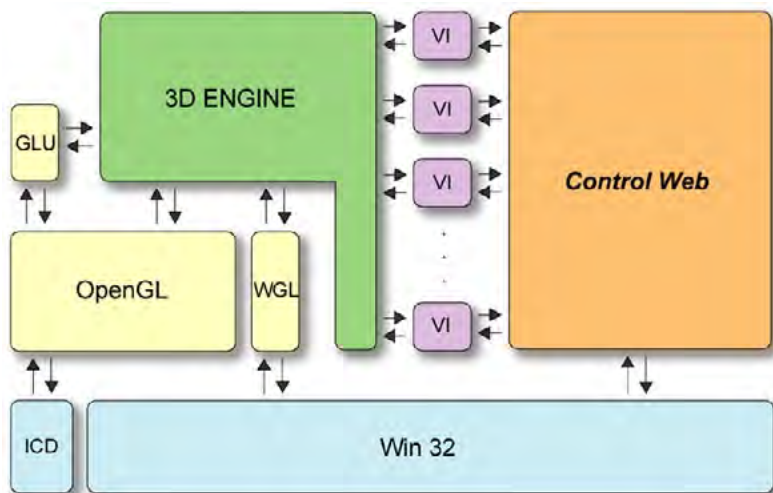
Obrázek 2 — Rozsah platformem a škálovatelnost systému Control Web

systemu, charakterizovaná především rozvojem internetových technologií a komunikací vůbec.

Zatímco Control Web 3.1 byl z převážné části replikou systému Control Panel 3.1 pro Windows, další systém Control Web 4, známý pod výročním pojmenováním **Control Web 2000**, přinesl intenzivní technický rozvoj. Koneckonců i nyní, po několika letech prodeje moderního systému Control Web 5, řada zákazníků stále žádá systém Control Web 2000. Tato verze přinesla možnosti přímé komunikace virtuálních přístrojů v počítačové síti, byla implementována rozhraní podle standardů OPC, ActiveX, ODBC, SQL a především do systému přibyla komponenta HTTP serveru s možností dynamického generování WWW stránek z běžícího aplikačního programu.

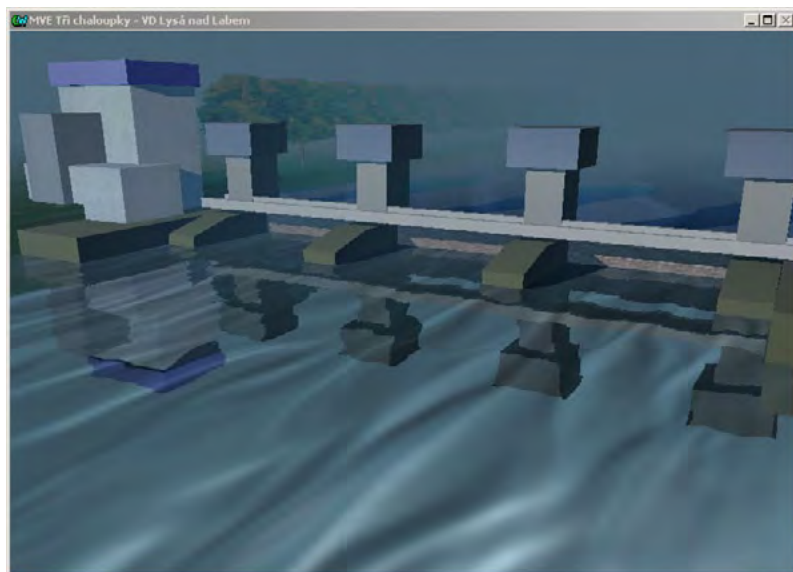


Obrázek 3 — Control Web může díky své škálovatelnosti vystupovat v mnoha rolích

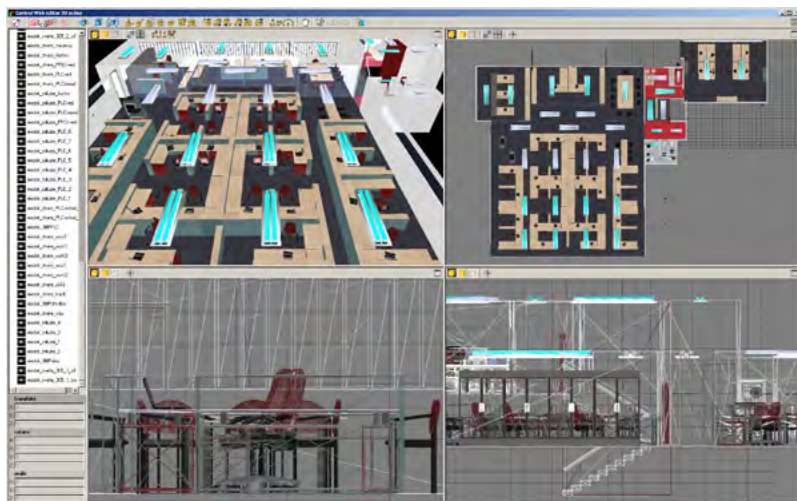


Obrázek 4 — Struktura začlenění 3D vykreslovacího serveru do prostředí Control Web

Control Web 2000 také přinesl možnost provozování aplikací v operačním systému Windows CE. Aplikace lze pohodlně vyvíjet na stolním PC s vývojovou verzí systému a poté z nich vygenerovat runtime pro veškeré platformy a používané procesory systému Windows CE. V jednom prostředí a stejným



Obrázek 5 — Obraz vytvořený 3D vykreslovacím systémem



Obrázek 6 — Editor 3D scény s několika otevřenými pohledy v prostředí Control Web

způsobem lze tedy na jedné straně vytvářet rozsáhlé aplikace pro cluster Windows Advanced Server a na straně druhé aplikace pro mobilní telefony a malé vestavěné počítače.

Asi nejrozsáhlejší změnou v historii vývoje systému byl příchod prostředí **Control Web 5** v roce 2004. Systém byl nyní schopen vytvořit skupinu redundantních aplikací bez podpory nákladného serverového systému, pouze prostřednictvím propojení v jedné počítačové síti. Nově byl zaveden koncept datových sekci, které v sobě zapouzdřují množinu datových elementů a umožňují nebo alespoň podstatně zjednodušují sdílení, zálohování a replikaci dat v síťovém prostředí. Řada vylepšení architektury zasahuje prakticky všechny části systému, mohutně je rozšířen také programovací jazyk OCL pro komunikaci mezi komponentami uvnitř aplikace.

Vizualizační schopnosti byly v páté verzi produktu rozšířeny o **třírozměrný vykreslovací systém**. Tento systém je postaven na principu klient – server. Vykreslovací server běží ve vlastním threadu a je schopen v reálném čase vykreslovat i velmi složité scény s mnoha efekty v téměř fotorealistické kvalitě. Využívá nejnovějších technologií počítačové grafiky, včetně shaderů běžících v grafických procesorech. Přitom, pokud není v aplikaci žádný 3D virtuální přístroj, nic z vykreslovacího systému není zaváděno do paměťového prostoru procesu. 3D systém tedy nemůže být na překážku ani u malých embedded systémů nevybavených možnostmi 3D grafiky. Tato nejdokonalejší současná technologie počítačové grafiky může podstatně zvýšit rychlost, působivost i přehlednost vizualizací.

Šestá verze systému v roce 2007 pak rozšířila paletu nabízených virtuálních přístrojů i nástrojů pro ladění aplikací, přinesla interní binární databázový

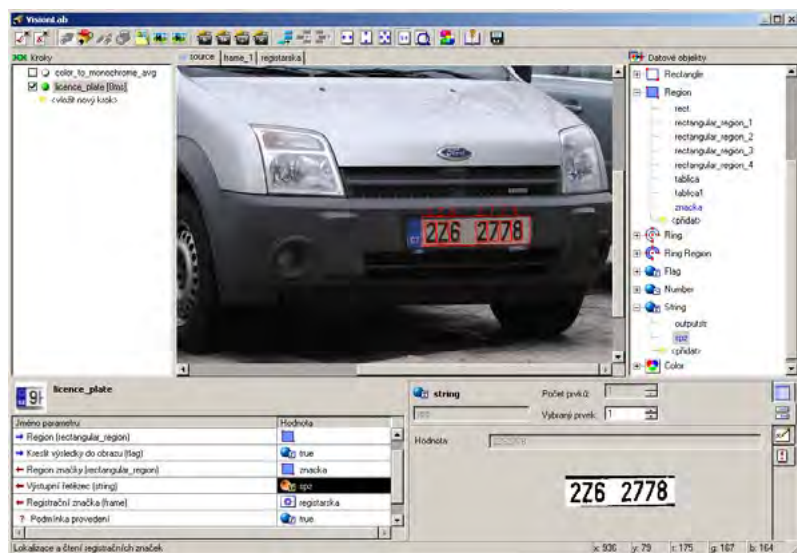
editor pro možnost práce bez SQL serveru. Rozšířila také možnosti vytváření 3D aplikací tím že součástí vývojového prostředí se stal **celoobrazovkový editor 3D scény**.

Control Web se v současné verzi rozrostl do podoby mohutného a velmi všestranného systému, který asi málokdo dokáže využít v jeho plné šíři. Technologie skládání aplikací z programových komponent, jejíž první nástin jsme přinesli již v roce 1992, se ukázala jako nosná a dodneška velmi silná. To ostatně potvrzuje nejen historie tohoto produktu, ale vůbec celý současný vývoj v oblasti softwarového inženýrství.

4. Vision Lab – strojové vidění pro průmyslové aplikace

Současné aplikace průmyslové automatizace nezřídko vyžadují automatizovat vizuální kontrolu – od vizuální inspekce práce automatizačních linek, přes zpracování snímaných obrazů, rozpoznávání tvarů, registračních značek aut, až po přesná měření. Proto byla vyvinut vývojový systém **Vision Lab** jako nadstavba systému Control Web pro strojové vidění

Aplikace strojového vidění měly dlouho pověst složitých, drahých a celkově nedostupných technologií. Ve většině těžkostí, které dříve realizaci těchto systémů komplikovaly a prodražovaly se situace hodně změnila. Výpočetní výkon běžných současných počítačů již umožňuje práci s velkými objemy obrazových dat v reálném čase. Také jsou již cenově velmi dostupné digitální kamery s dostatečně kvalitním obrazem. Klíčovým faktorem při realizaci systémů strojového vidění je nyní především programové vybavení.



Obrázek 7 — Editor aplikací strojového vidění Vision Lab ve vývojovém prostředí Control Web

A nejedná se zde pouze o jeho výkonnost, rozsah funkčnosti a spolehlivost, ale v neposlední řadě také o snadnost používání.

System VisionLab je vstřícný i k méně zkušeným uživatelům a je navržen tak, aby co nejvíce usnadnil sestavení úlohy strojového vidění. Vývojové prostředí je integrováno do systému Control Web a je velmi intuitivní a interaktivní. Jak je v systému Control Web zvykem, také veškeré operace s obrazem lze vybírat z nabídkových palet, zařazovat do aplikace a také lze okamžitě pozorovat výsledky těchto operací.

VisionLab aplikacím přináší:

- snadnou integraci digitálních obrazů a vizuální inspekce do aplikací v průmyslové automatizaci
- velký výběr výkonných a technicky vyspělých kroků pro práci s obrazem
- intuitivní editaci kroků řetězce strojového vidění
- podporu plně paralelního zpracování na více jádrech a více procesorech
- podporu masivně paralelního zpracování obrazu grafickým procesorem
- pokročilé úpravy obrazu prováděné grafickým procesorem
- přenos obrazových dat v počítačových sítích
- archivaci obrazových dat v podobě snímků i videosouborů
- otevřené rozhraní pro doplňování kroků strojového vidění
- sdílení dat s aplikacemi systému Control Web
- snadnou integraci aplikací strojového vidění a vizuální inspekce do větších informačních a řídicích systémů
- plug and play instalace ovladačů kamer DataCam
- možnost používat veškeré kamery s instalovanými WDM ovladači, tedy např. i běžné webové kamery a veškeré kamery, které jsou součástí notebooků

Autor aplikace má k dispozici velký výběr samostatných kroků, které může snadno zařazovat do řetězce zpracování a vyhodnocování obrazu z kamery.

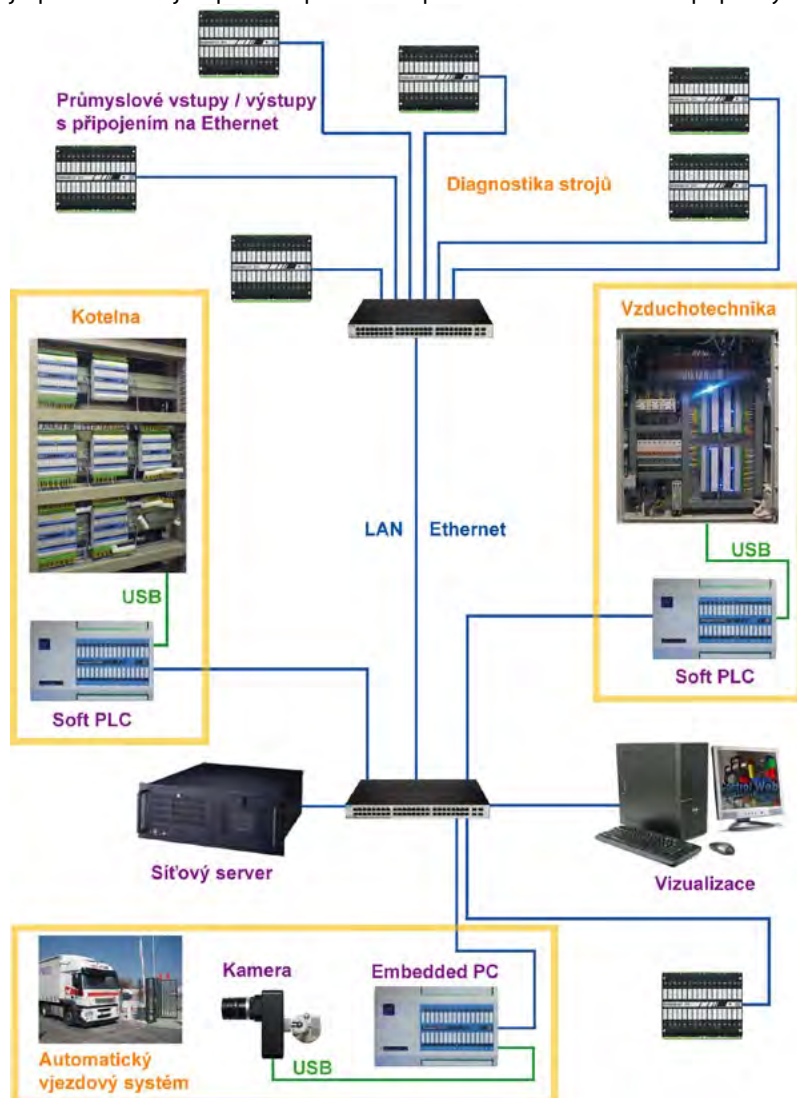
Algoritmy uvnitř kroků zpracování obrazu vždy maximálně využívají možností počítače, na kterém běží. U vícejádrových centrálních procesorů je výpočet rozdělen paralelně na všechna jádra. V systému jsou k dispozici i kroky, které dokážou využívat i mohutného masivně paralelního výkonu současných programovatelných grafických procesorů. Zpracování obrazu pak může probíhat současně i v několika stovkách paralelních větvích. Uživatelská přívětivost vývojového prostředí nikterak negativně neovlivňuje vysoký výkon spuštěných aplikačních programů.

Rozhraní pro kroky je otevřené a algoritmy zpracování obrazu tak lze neomezeně doplňovat.

Vstupní data pro řetězec zpracování obrazu mohou být naplněna aplikačním programem v prostředí systému Control Web a výstupní data jsou po proběhnutí řetězce v tomto prostředí opět k dispozici. Tak je velice usnadněna integrace úloh strojového vidění a vizuální inspekce do vyšších propojených celků. Úloha strojového vidění tak může např. přímo řídit stroj, spolupracovat s SQL databázemi, využívat HTTP servery, posílat SMS, komunikovat s jinými úlohami v síti atd.

5. Současnost – neomezený prostor pro technologické aplikace

Control Web vždy byl a dodnes je cenově atraktivní, proto je používán nejen v rozsáhlých aplikacích ve velkých firmách, ale i v malých a vestavěných aplikacích a také ve školách, ve vědě a výzkumu [1]. Struktura typických aplikací se v poslední době podstatně změnila. Postupně ubývá zakázek, ve kterých je požadováno jen prosté operátorské pracoviště s vizualizací a případným



Obrázek 8 — Control Web jako integrující prvek komplexních systémů

sběrem dat. Požadavek rozhraní pro webové klienty je již samozřejmostí. Většina současných systémů je zapojena, často bezdrátově, do počítačových sítí, obvyklá je spolupráce s nějakým databázovým informačním systémem. Také se často systémy skládají z více částí, které spolu musejí komunikovat. Stále větší výhodou se tak stává to, že i malý vestavný systém je vybaven veškerými komunikačními kanály a jeho programové vybavení dokáže pracovat se všemi současnými standardy výměny dat. I malá vestavěná řídicí jednotka tak má k dispozici Ethernet, USB, Wi-Fi, Bluetooth a může obsahovat internetový HTTP server, ale současně také webový klient, dokáže posílat e-maily, posílat a přijímat SMS zprávy, komunikovat přes GPRS nebo radiové mosty, spolupracovat s Plug-and-Play zařízeními na rychlé USB atd. V řadě případů dokáže malý a levný průmyslový počítač nahradit kombinaci PLC a počítače pro operátorské řízení. Zde pak výhoda jednoho programového prostředí pro vývoj aplikací nabývá na ceně.

Control Web je programovým systémem, který dokáže vystupovat v mnoha rolích. Může pracovat v řídicích jednotkách strojů, může spojovat výrobní technologii s informačním systémem podniku, může být datovým serverem s mnoha webovými klienty, může modelovat a simulovat procesy, dokáže vytvářet náročné vizualizace a mnoho dalšího.

Velká perspektiva se otevírá i při **biotechnologických aplikacích**. Může řídit biotechnologický automat či robot, snímat a vyhodnocovat fotometrická měření či strojově zpracovávat nasnímaný obraz, pro přenos naměřených dat se propojit přes internet nebo lokální síť s informačním systémem nemocnice apod.

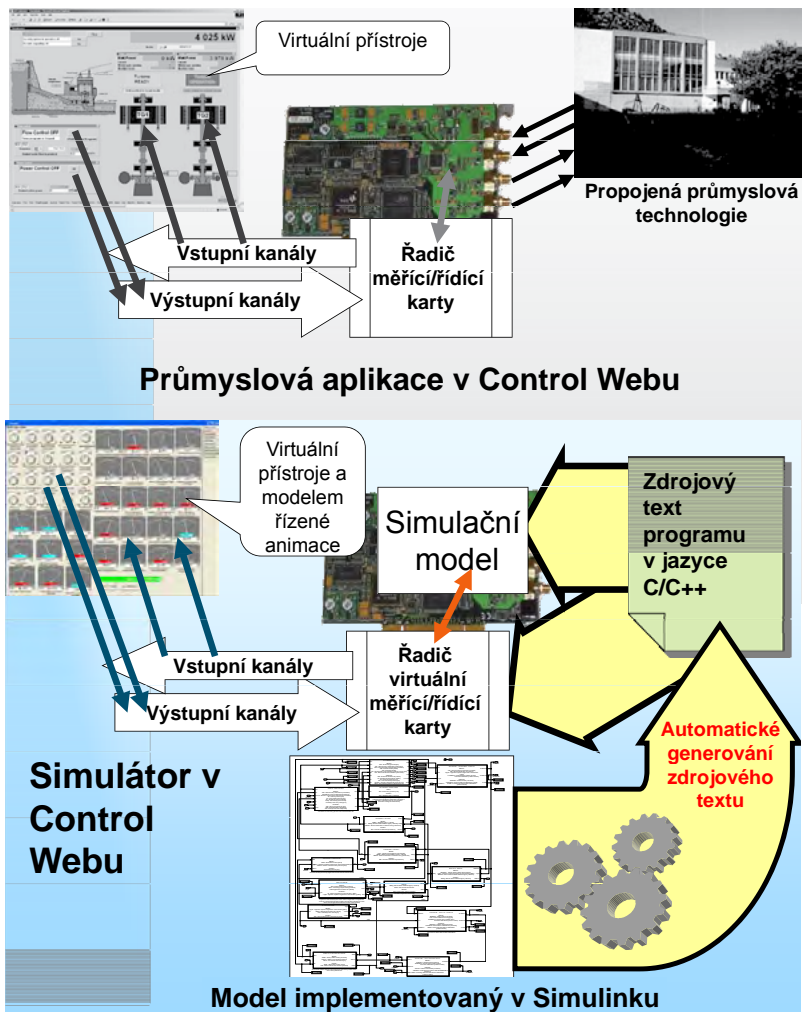
6. Control Web pro simulace

Control Web poskytuje velmi výkonné prostředky pro vývoj uživatelského rozhraní. Toho jsme se snažili využít při tvorbě výukových simulátorů. V Control Webu byl mimo jiné realizován simulátor GOLEM [2].

Základními stavebními kameny uživatelské aplikace jsou tzv. virtuální přístroje (multimediální komponenty imitující skutečné součásti průmyslových velinů – nejrůznější měřicí přístroje, grafy, přepínače apod.). Virtuální přístroje reagují na změnu hodnot specifikovaných proměnných (např. posunem ručičky měřicího přístroje, zobrazením hodnoty na displeji, rozsvícením žárovky, vykreslením grafu apod.), nebo naopak, po posunu ovládacího knoflíku či stisku přepínače (prostřednictvím počítačové myši) virtuální přístroj změni hodnotu příslušné proměnné.

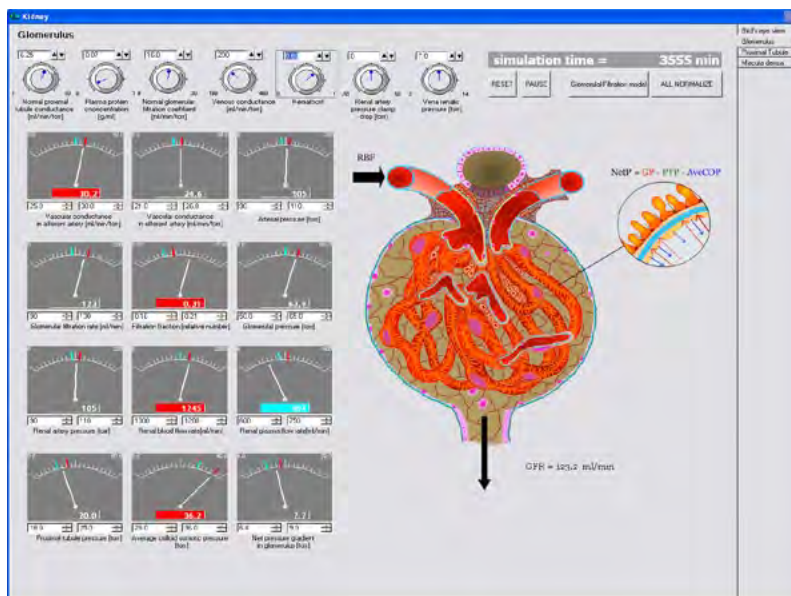
Měřené hodnoty okolního světa jsou virtuálním přístrojům zprostředkovány přes vstupní kanály (a virtuální přístroje na tyto hodnoty příslušným způsobem reagují), řídicí signály do okolí mohou virtuální přístroje posílat pomocí výstupních kanálů.

Abychom mohli využít vývojářské pohodlí systému Control Web při tvorbě simulátorů, bylo nutno napsat speciální ovladač, který byl schopen komunikovat (přes softwarové kanály) s objekty systému Control Web.



Obrázek 9 — Virtuální přístroje v průmyslové aplikaci v Control Webu komunikují prostřednictvím vstupních a výstupních kanálů přes řadič řídicí/měřicí karty s propojenou technologií, v simulátoru komunikují se simulačním modelem implementovaným jako ovladač virtuální měřicí/řídící karty.

Na rozdíl od ovladačů ke skutečným měřicím a řídicím kartám však tento ovladač nekomunikuje s hardwarem těchto karet, ale se simulačním modelem, který je součástí ovladače. Pokud se ovladač napíše dobře, je systém Control Web „ošálen“: vstupní kanály (k měřicím přístrojům) považuje za skutečné měřené signály někde v technologickém okolí počítače, zatímco ve skutečnosti to jsou výstupní proměnné simulačního modelu. Výstupní kanály, odcházející od



Obrázek 10 — vizuální rozhraní ve výukovém simulátoru funkce ledvin vytvořeného v prostředí Control Web. Výstupy modelu jsou zobrazovány na ručkových měřících přístrojích a zároveň ovlivňují i tvar animovaného obrázku ledvinného glomerulu (průměry cév, tloušťku šipek a číselnou hodnotu aj.), vytvořeného pomocí programu Adobe Flash.

řídících prvků systému Control Web, nenastavují přes příslušný ovladač nějaké aktivní prvky technologie, ale mění vstupy simulačního modelu. Vzájemný vztah mezi simulačním modelem ve virtuálním ovladači a vizualizačním rozhraním systému Control Web je realizován jako klient – server technologie: v určitých časových okamžicích Control Web (klient) žádá virtuální ovladač se simulačním modelem (server) o načtení hodnot ze vstupních kanálů a umožnění změny hodnot výstupních kanálů.

Jádrom simulátoru je tedy virtuální ovladač, který obsahuje vlastní simulační model. Modelové jádro simulátoru jako ovladač virtuální měřící/řídící karty, bylo zprvu implementováno v jazyce Modula a později v C++.

Pro implementaci ovladače jsme využívali simulační model, o jehož adekvátnosti jsme se (se vším pohodlím systému MATLAB a SIMULINK) již přesvědčili.

Při změnách matematického modelu, odladovaného v prostředí Matlab/Simulink (tj. v prostředí pro vývoj matematických modelů) bylo ale nutné tyto změny vždy promítnout i do simulačního jádra simulátoru implementovaného v C++. Proto jsme postupně hledali cesty jak tuto transformaci usnadnit a nemuset tento ovladač psát v C++ „ručně“. Vyvinuli jsme proto speciálního nástroj, který umožní vývoj ovladače automatizovat. To nám umožnilo

ze simulinkového schématu přímo generovat zdrojový text příslušného virtuálního ovladače v C++. Tím bylo možné jednoduše modifikovat ovladač v prostředí Control Web při nejrůznějších úpravách simulačního modelu v prostředí Simulink [2].

Nyní pracujeme na tvorbě obdobného nástroje, který by nám umožnil generovat ovladač z modelu vytvořeného v prostředí jazyka Modelica.

Literatura:

- [1.] Cagaš, R.: *Control Web – všestranný pomocník pro tvůrce aplikačních programů.* AUTOMA, 6/2011 str. 1–4
- [2.] Kofránek, J., Anh Vu, L. D., Snášelová, H., Kerekeš, R., & Velan, T.: *GOLEM – Multimedia simulator for medical education.* V L. Patel, R. Rogers, & R. Haux (Editor), *Studies in Health Technology and Informatics, 2001*, vol. 84, MEDINFO 2001, *Proceedings of the 10th World Congress on Medical Informatics.* 1042-1046. London: IOS Press, 2001. Práce je dostupná na adrese <http://www.physiome.cz/references/MEDINFO2001.pdf>

Kontakt:

Ing. Roman Cagaš

Ing. Pavel Cagaš

Moravské přístroje a.s.,

Masarykova 1148

763 02 Zlín - Malenovice

e-mail: rc@mii.cz

<http://mii.cz>

doc. MUDr. Jiří Kofránek, CSc.,

Oddělení biokybernetiky,

Ústav patologické fyziologie 1. LF UK

U nemocnice 5, 128 53 Praha 2

tel: +420 777-686868

e-mail: kofranek@gmail.com

<http://physiome.cz>