

# CONTROL WEB PRO MULTIMEDIÁLNÍ INTERAKTIVNÍ LEDVINU

*Jiří Kofránek, Martin Tribula*

## **Anotace**

Vytvořili jsme výukový model základní funkce ledvin, který byl vyvinut v prostředí Matlab/Simulink. Vytvořili jsme speciální softwarový nástroj, který ze simulačního modelu z prostředí Matlab/Simulink vygeneruje řadič virtuální řídicí karty s níž je schopen komunikovat systém Control Web, původně určený pro vývoj průmyslových řídicích a vizualizačních aplikací. Control Web je tak "ošálen" – prostřednictvím řadiče virtuální řídicí karty "se domnívá", že čte data z průmyslového zařízení ale ve skutečnosti čte výstupy simulačního modelu. Obdobně, Control Web přes tento řadič posílá signály do průmyslového zařízení - ve skutečnosti posílá vstupy do simulačního modelu. To nám dalo možnost jednoduše a rychle vytvořit uživatelské rozhraní výukového modelu v prostředí Control Web. Vlastní výuková aplikace vyžaduje pro svůj běh speciálně upravené volně šiřitelné runtime prostředí systému Control Web, které lze, spolu s vlastním modelem, nainstalovat z Internetu.

## **Klíčová slova**

*E-learning, krevní plyny, ledviny, simulační model, výukové simulátory*

## **1. Úvod**

Ledviny jsou orgánem, který reguluje vylučování celé škály látek a tím i jejich hladiny ve vnitřním prostředí. Ledviny jsou velmi složitě regulovaný dynamický systém. Ovlivňují objem cirkulující krve a tím i hemodynamické parametry. Na druhé straně změny cirkulace mají svou odezvu v činnosti ledvin. Pochopení funkce jednotlivých částí ledvin je důležité např. pro posouzení poruch jejich funkce a vedlejších projevů diuretické léčby. Simulační hry jsou jedním z účinných prostředků, usnadňujících pochopení dynamiky složitých regulačních vztahů v ledvinách. Tvorba e-learningového programu, využívajících simulační hry s virtuálními (zdravými i nemocnými) ledvinami byla i naším úkolem.

## **2. Na začátku je matematický model**

Simulační modely ledvin jsou jedním z nástrojů interpretace experimentálních dat. V odborné literatuře byla popsána celá řada

modelů ledvin, od jednoduchých až po velmi složité modely, explicitně zahrnující i model miliónu nefronů, řešených na superpočítači. Pro účely výukových aplikací jsme zvolili spíše modely se soustředěnými parametry, jejichž účelem je jejich využití pro vysvětlení obtížněji pochopitelných partií fyziologie a patofyziologie ledvin. Konkrétně se jedná o modelování funkce autoregulace glomerulární filtrace a průtoku krve ledvinami. Modelování tubuloglomerulární vazby a její adaptace na dlouhodobé podněty. Modelování funkce macula densa a smyčky renin-angiotenzin-aldosteron. Modelování funkce protiproudového systému a Henleho kličky a vlivu hemodynamiky na funkci protiproudového systému. Modelování vlivu antidiuretického hormonu. Modelování regulace vylučování draslíku a sodíku. Modelování renální regulace acidobazické rovnováhy.

Modely vytváříme v prostředí **Matlab-Simulink**. Jedná se v podstatě o několik modelů /na různé úrovni složitosti/, které mohou být využívány v různých simulačních hrách. Rovněž uvažujeme i malé modely ledvinných subsystémů, které mohou být studovány odděleně (po rozpojení příslušných regulačních smyček), pro lepší pochopení funkce vybraného subsystému. Tak je např. možno studovat vliv rezistencí přívodní a odvodní arterioly, hladiny plazmatických bílkovin, permeability glomerulární membrány i protitlaku proximálního tubulu na velikost glomerulární filtrace.

Model je ovšem pouze začátek. Pro výukovou simulační aplikaci ho musíme **převést z prostředí Matlab/Simulink do prostředí vytvářeného výukového simulátoru** a názorně při tom vizualizovat jeho chování.

### 3. Vizualizační prostředí pro simulátory

Od simulátoru vyžadujeme velké možnosti vizualizace vstupů a výstupů modelu (ve formě nejrůznějších přehledných displejů, grafů, či modelem řízených pohyblivých schémat a animací). Na druhé straně vlastní model nezdědka bývá výpočetně poměrně náročný. Od nástroje, v němž budeme vytvářet vlastní výukový simulátor, tedy požadujeme poměrně protichůdné požadavky: na jedné straně chceme aby podporoval velké **možnosti grafické vizualizace** a na druhé straně vyžadujeme zachování dostatečně velké **kapacity procesoru pro souběžné numerické výpočty**.

Obdobné nároky mají systémy pro návrh **průmyslových řídicích a měřících systémů** (při vykreslování výsledků vizualizace na obrazovce se nesmí zastavit paralelně běžící výpočet ovládající nějaký průmyslový proces). Proto jsme se při hledání vhodného nástroje pro tvorbu

vizualizací výukových simulačních her poohlédli po nástrojích původně určených pro vizualizaci průmyslových procesů.

Nástrojů pro design průmyslových aplikací je na světovém trhu nemálo. Jejich ceny jsou ovšem, na rozdíl od obecných softwarových nástrojů, zpravidla velmi vysoké. Z těchto (a dalších) důvodů jsme se poohlédli po domácím výrobcí, kterým nakonec byla zlínská akciová společnost "**Moravské přístroje,**" která vyvíjí systém pro tvorbu průmyslových aplikací s názvem "**Control Web**".

#### 4. Virtuální součástky pro virtuální ledvinu

"Svensk kvalitet (till Tjeckiska priser)" – "Švédská kvalita (za české ceny)", stojí v letáčku skandinávského distributora výkonného softwarového nástroje z Valašska. Švédové jsou velmi domýšliví na kvalitu svých výrobků, a pokud o vývojovém nástroji **Control Web** z dílny zlínské společnosti Moravské přístroje veřejně prohlašují, že dosahuje "švédské kvality", je to velké ocenění skupiny tvůrců, kteří na tomto produktu v potu tváře usilovně pracují již od počátku devadesátých let. Nyní je jejich úsilí korunováno tím, že jejich programové dílo (jehož zdrojový text obsahuje více než 2 milióny řádek) nachází úspěšnou cestu na zahraniční trhy. Poslední verze programu je tak možno vidět v nejen v anglické, německé či švédské verzi, ale i v "rozsypaném čaji" japonských znaků.

**Control Web** je především určen pro **vývoj průmyslových vizualizačních a řídicích aplikací na platformě WIN32** - sběr, ukládání a vyhodnocování dat, tvorba rozhraní člověk-stroj aj. (viz <http://www.mii.cz>). Objektově-orientovaná komponentová architektura zajišťuje vyvíjeným aplikacím široký rozsah nasazení: od prostých časově nenáročných vizualizací až po řídicí aplikace reálného času, od jednoprosesorových aplikací až po rozsáhlé distribuované síťové aplikace.

V průmyslových aplikacích jádro systému Control Web komunikuje přes ovladač měřicí/řídicí karty s průmyslovým zařízením. Na jádro systému je napojena uživatelská aplikace vytvořená pomocí vizualizačních nástrojů.

Základními stavebními kameny uživatelské aplikace v Control Webu jsou **virtuální přístroje** (komunikující mezi sebou pomocí proměnných a zpráv). Virtuální přístroje představují širokou paletu zobrazovacích nástrojů - od ručkových měřících přístrojů, přes nejrůznější škálu grafů až po nástroje pro archivaci zobrazovaných dat a "hlídací" komponenty, spouštějící reakci na odchylku měřených dat od definovaného stavu.

Každý virtuální přístroj je komponenta, jejíž vlastnosti jsou pomocí vizualizačních a ovládacích prvků snadno modifikovatelné. Ale nejenom to – každá komponenta systému má k dispozici mocné programátorské nástroje, jako jsou lokální proměnné a libovolně definovatelné procedury, reagující na události.

Pro rychlý vývoj uživatelského rozhraní Control Web poskytuje velmi efektivní prostředky. Tak např. z palety virtuálních přístrojů je možno snadno tažením myši vytáhnout potřebný přístroj a umístit ho na příslušný panel. V interaktivním dialogu je pak možné virtuálnímu přístroji nastavit hodnoty jeho příslušných atributů, napojit ho na příslušný "vstupní kanál" ze kterého bude odečítat nějaké hodnoty, nebo naopak mu vybrat "výstupní kanál" do kterého bude nějaké nastavované hodnoty posílat, nadefinovat lokální proměnné přístroje, či individuální procedury přístroje (metody objektu) apod.

Výhodnou vlastností Control Webu je to, že jedním z virtuálních přístrojů je **kontejner pro komponenty ActiveX**, který tak představuje most mezi systémem Control Web a vlastnostmi a metodami (OLE Automation) ActiveX komponent. To znamená, že do aplikace lze zabudovat ActiveX komponenty a programově je ovládat - nastavovat jim vlastnosti a volat metody z procedur jakýchkoliv přístrojů. Můžeme tedy pomocí programu **Adobe Flash MX** vytvořit interaktivní výukové animační obrázky a ty pak uložit jako ActiveX komponenty do kontejneru. Animace pak mohou být řízeny na základě výstupů implementovaného simulačního modelu – např. na schematicém obrázku ledvinného glomerulu se cévy se mohou roztahovat nebo komprimovat, tloušťka šipky znázorňující velikost glomerulární filtrace se může ztlušťovat či ztenčovat atd.

Celá výuková aplikace v Control Webu pak může vypadat jako sada propojených interaktivních obrázků na jejichž pozadí může běžet dosti komplikovaný simulační model.

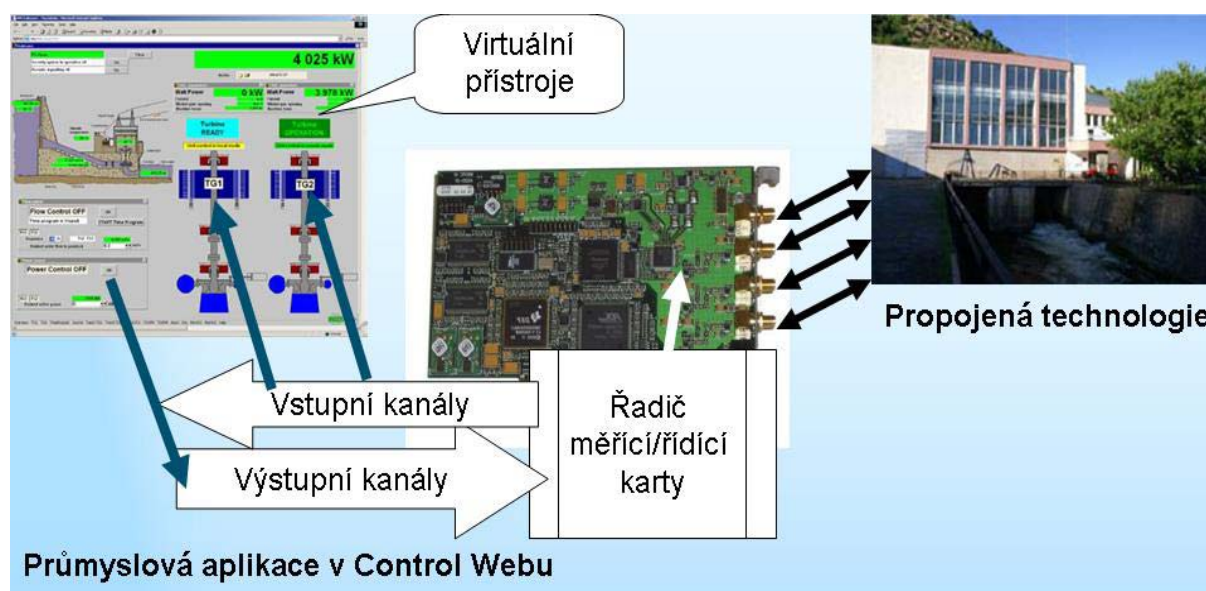
## 5. Jak ošálit Control Web

**Měřené hodnoty** okolního světa jsou v Control Webu virtuálním přístrojům zprostředkovány přes **vstupní kanály, řídicí signály** do okolí mohou virtuální přístroje posílat pomocí **výstupních kanálů**.

Vstupní a výstupní kanály komunikují s ovladači příslušných měřících/řídicích karet, které jsou propojeny s vnějším technologickým zařízením. Z něj pak do karet přicházejí měřené signály, které ovladač přetvoří do příslušných numerických hodnot vstupních kanálů systému ControlWeb, kde jsou vizualizovány pomocí virtuálních přístrojů napojených na tyto kanály. Naopak, nastavované hodnoty ve výstupních

kanálech jsou ovladačem a řídicí kartou přetvořeny na příslušné řídicí signály, které jsou posílány do připojené technologie.

Abychom mohli využít vývojářské pohodlí systému Control Web při tvorbě simulátorů, musíme použít určitý "úskok". Je nutno v jazyce C++ naprogramovat **speciální ovladač**, který je schopen komunikovat (přes softwarové kanály) s objekty systému Control Web. 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. Systém Control Web je tak "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 řídicích prvků systému Control Web, nenastavují přes příslušný ovladač nějaké aktivní prvky technologie, ale mění vstupy **simulačního modelu**.



**Obr. 1** Virtuální přístroje v průmyslové aplikaci v Control Webu komunikují prostřednictvím vstupních a výstupních kanálů přes řadič řídicí/měřicí karty s propojenou technologií

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.

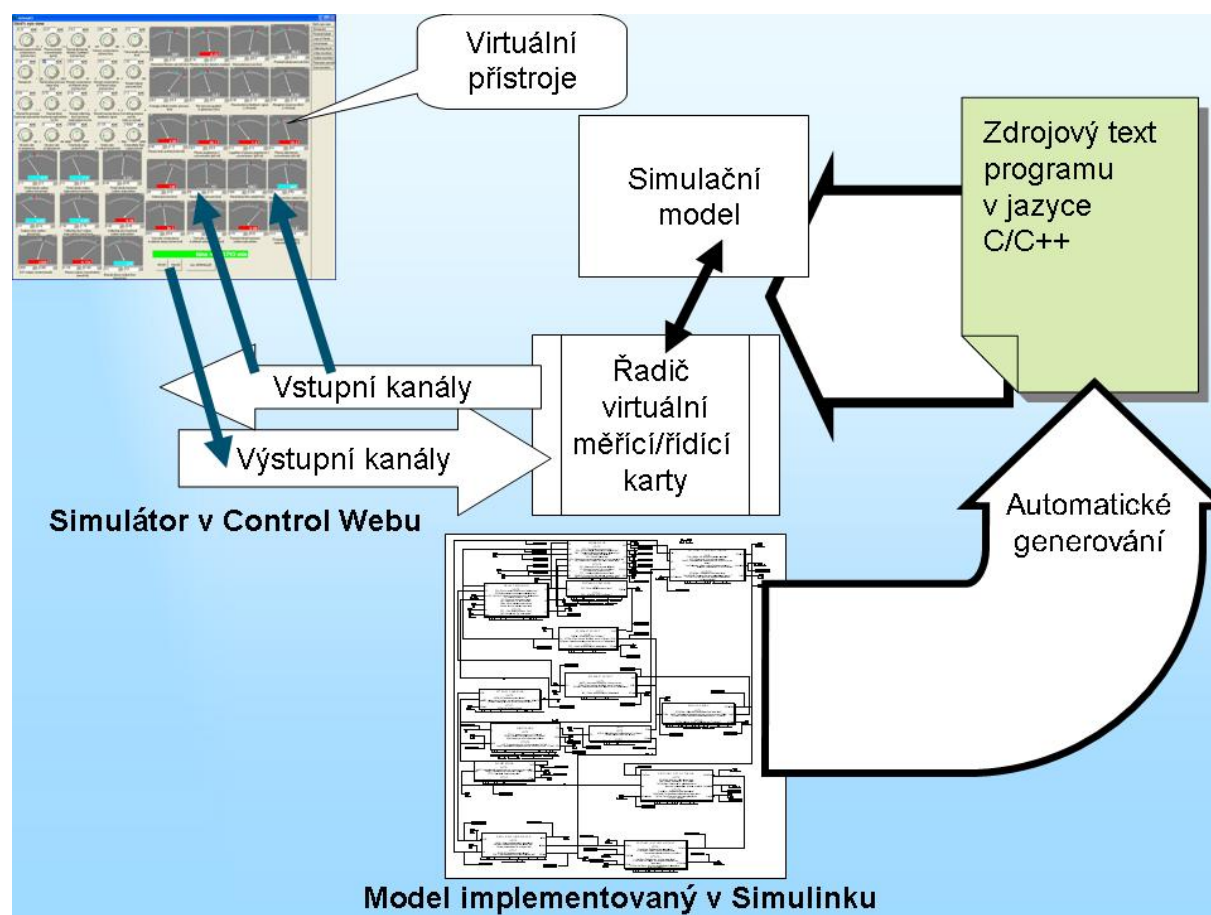
Vytvořený simulátor vyžaduje, aby na počítači, kde má simulátor běžet byl nainstalován runtime prostředí Control Web. Pro distribuci simulátorů prostřednictvím Internetu (včetně potřebného runtime

prostředí Control Web) využíváme speciální program "dispečer simulačních modelů", o kterém jsem referovali na předminulém Medsoftu [4].

## 6. Dřinu strojů!

Jádrum simulátoru je **ovladač virtuální řídicí/měřicí karty**, který obsahuje vlastní simulační model. Ovladač je implementován v jazyce C nebo C++ nebo Modula 2. Jako podklad pro tvorbu využíváme simulační model, o jehož adekvátnosti jsme se (se vším pohodlím systému Matlab/Simulink) již přesvědčili.

Pomocí Control Webu a speciálního ovladače s vnořeným simulačním modelem (psaným v Module 2 a později v C++) jsme např. v minulosti realizovali implementaci simulátoru fyziologických regulací simulátoru fyziologických funkcí Golem [2].



**Obr. 2** Virtuální přístroje v simulátoru vytvořeném v Control Webu komunikují prostřednictvím vstupních a výstupních kanálů přes řadič (neexistující) virtuální řídicí/měřicí karty se simulačním modelem, který je součástí řadiče. Zdrojový text programu tohoto řadiče je možné automaticky vygenerovat z modelu, implementovaném v Simulinku pomocí námi vytvořeného softwarového nástroje.

Je ale nutno poznamenat, že programování obdobných speciálních ovladačů pro systém Control Web je poměrně časově náročná a úmorná mravenčí práce. Krom toho, při jakýchkoli změnách v modelu (který vytváříme v pohodlí vývojového systému Matlab/Simulink) je pak nutno přeprogramovat celý ovladač virtuální karty.

Abychom vývoj ovladačů (a vnořeným simulačním modelem) usnadnili a nemuseli ho psát v C++ "ručně", vyvinuli jsme speciální program, který **umožní vývoj ovladače automatizovat**.

Díky tomu máme nyní možnost ze simulinkového schématu přímo generovat zdrojový text příslušného virtuálního ovladače v jazyce C.

To nám nyní dává možnost **oddělit tvorbu simulačního modelu a tvorbu simulátoru**. Simulační model vytváříme ve vývojovém prostředí, určeném pro vývoj simulačních modelů – tj. v prostředí Matlab/Simulink, a pro vývoj vlastního simulátoru využíváme širokou škálu vizualizačních nástrojů, které nabízí vývojové prostředí Control Web. Propojení mezi oběma prostředími zajišťuje námi vyvinutý softwarový nástroj, který převede model vytvořený v Simulinku do podoby ovladače virtuální karty pro Control Web.

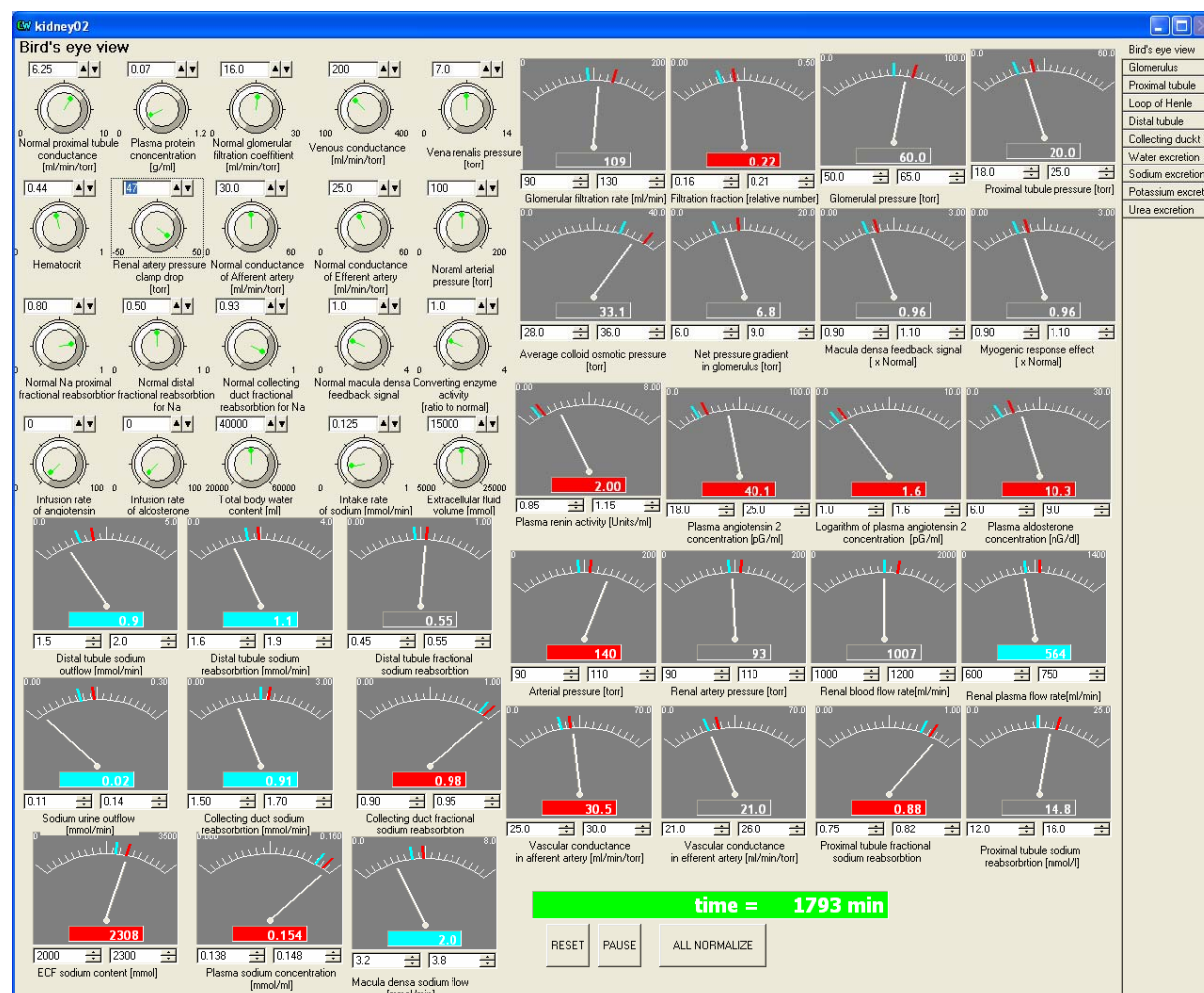
Nyní je snadné při nejrůznějších úpravách simulačního modelu v prostředí Simulink jednoduše modifikovat ovladač v prostředí Control Web a vytvořit novou verzi ovladače s novou verzí simulačního modelu.

## 7. Od umění k průmyslu (při tvorbě simulátorů)

Zdá se, že pomalu končí doba, kdy vytváření výukových programů bylo otázkou entuziasmu a píle skupin nadšenců. Tvorba moderních výukových aplikací je poměrně náročný a komplikovaný projekt, vyžadující týmovou spolupráci řady profesí – od zkušených učitelů, jejichž scénář je základem kvalitní výukové aplikace, přes systémové analytiky, kteří ve spolupráci s profesionály daného oboru jsou odpovědní za vytvoření simulačních modelů pro výukové simulační hry, výtvarníky, kteří vytvářejí vnější vizuální podobu, až po programátory, kteří celou aplikaci "sešijí" do výsledné podoby. Aby tato interdisciplinární kolektivní tvorba byla efektivní, je nutno pro každou etapu tvorby využívat specifické vývojové nástroje, s dostatečnou technickou podporou, které umožňují komponentovou tvorbu simulačních modelů, vytváření interaktivních multimédií a jejich závěrečné propojení podle daného scénáře do kompaktního celku [3].

Při tvorbě simulátorů v naší laboratoři proto využíváme Matlab/Simulink pro návrh a odladění vlastního simulačního modelu, Control Web a Visual Studio .NET pro tvorbu simulátoru (pro automatizaci převodu modelu z prostředí Matlab/Simulink do podoby assembly v prostředí Microsoft .NET jsme referovali na minulém

Medsoftu [5]). Pro vytváření multimediálních interaktivních komponent uživatelského rozhraní využíváme Adobe Flash, z výsledné interaktivní animace pak prostřednictvím ActiveX komponent propojujeme se simulačním modelem (a to jak v simulátorech vytvořených pomocí Microsoft Visual Studia, tak i v simulátorech vytvořených v Control Webu).



**Obr. 3** Prvotní design simulátoru ledvin má příliš technický ráz

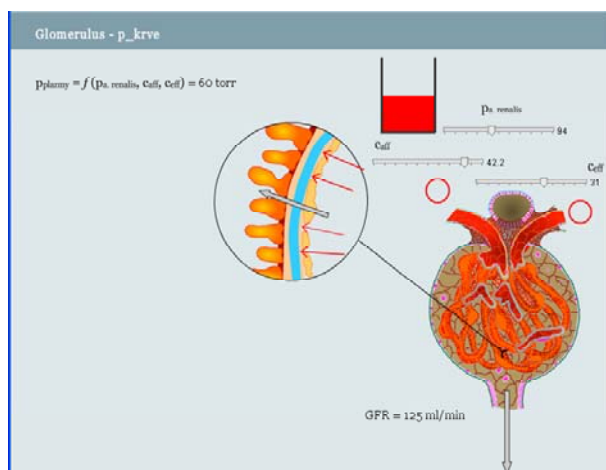
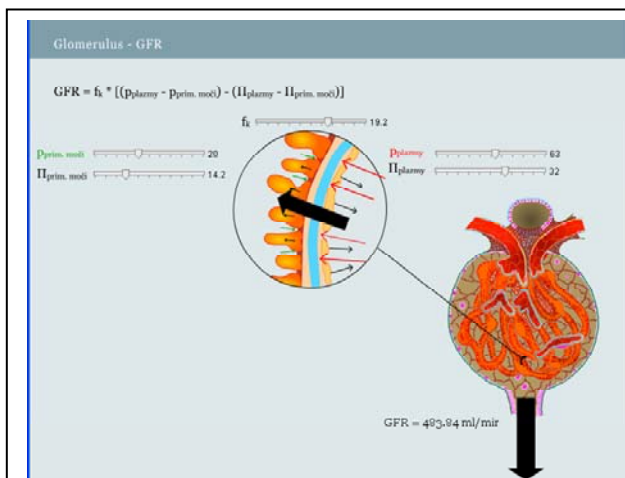
## 8. Ledvina na síti

Simulátor ledvin je součástí internetového atlasu fyziologických modelů [1], dostupném na [www.physiome.cz](http://www.physiome.cz), který dlouhodobě vytváříme v naší laboratoři.

Simulátor je určen studentům medicíny a lékařům pro studium fyziologie a patofyziologie ledvin. Má pomoci pochopit základní funkce ledvin a kauzální řetězce jejich poruch. Pro usnadnění výkladu model umožňuje rozpojit některé regulační smyčky s sledovat chování jednotlivých částí ledvin odděleně. Tak je např. možné sledovat faktory,



kteřé ovlivňují glomerulární filtraci a průtok krve ledvinami, modelovat rozpojenou a propojenou tubuloglomerulární vazbu a sledovat její



**Obr 4** Technický háv simulátoru (připomínající spíše průmyslový velín než výukový simulátor) mohou setřít interaktivní animace, vytvořené pomocí Adobe Flash, implementované do simulátoru pomocí ActiveX kontejneru a propojené se simulačním modelem

postupnou adaptaci na dlouhodobé podněty. Postupným rozpojováním a zapojováním regulačních smyček je možné podrobně sledovat funkci protiproudového systému ledvin, vliv různých koncentrací sodíku a draslíku k tubulární moči a v krvi na chování distálního nefronu při různých hladinách aldosteronu, chování ledvin při různých poruchách acidobazické rovnováhy apod.

## 8. Závěr

Vytvořený simulátor ledviny v Control Webu je první implementací s využitím nově vytvořeného nástroje umožňujícího automatizovaný převod vytvořeného modelu z prostředí Matlab/Simulink do vývojového prostředí Control Web. Na rozdíl od simulátorů, vytvořených v platformě Microsoft Studio .NET, je nutno na počítači, kde má být simulátor provozován, instalovat speciální volně šiřitelný runtime pro systém

Control Web. Rychlost vytváření aplikace je však v prostředí Control Web nesrovnatelně rychlejší – proto se tento systém hodí především pro vytváření prvotních prototypů simulátoru. Prvotní design aplikace vytvořené v Control Webu (s využitím integrovaných vizualizačních nástrojů) je na první pohled příliš technický. Nicméně, tak jak do ActiveX kontejnerů instalujeme interaktivní multimediální elementy vytvořené v Adobe Flash, technický háv uživatelského rozhraní aplikace se ztrácí a postupně se přeměňuje na obrázky interaktivního počítačového atlasu fyziologie, který je bližší těm, komu je aplikace určena – studentům medicíny a lékařům.

## 9. Literatura

- [1] Andrlík M., Kofránek J., Matoušek S. Stodulka P., Wünsch Z., Kripner T., Hlaváček J.: Internetový atlas multimediálních modelů pro vybrané kapitoly normální a patologické fyziologie člověka. Ukázka předběžných výsledků. Sborník příspěvků MEDSOFT 2006. ISBN 80-86742-12-1, str. 7-12, Zeithamlová M., Agentura Action-M, Praha 2006.
- [2] Kofránek, J., Anh Vu, L. D., Snášelová, H., Kerekeš, R. Velan, T. (2001): "GOLEM – Multimedia simulator for medical education." In: Studies in Health Technology and Informatics., vol. 84. MEDINFO 2001, Proceedings of the 10th World Congress on Medical Informatics. (V.L. Patel, R. Rogers, R. Haux (Eds.)), IOS Press, Amsterdam, Berlin, Oxford, Washington DC, 2001, pp. 1042-1046.
- [3] Kofránek J, Andrlík M, Kripner T, Stodulka P. From Art to Industry: Development of Biomedical Simulators. The IPSI BgD Transactions on Advanced Research 1 #2, 62-67. 2005. New York, Frankfurt, Tokyo, Belgrad. 2005.
- [4] Kofránek J., Andrlík M., Kripner T., Hlaváček J., Stodulka P.: E-learning s výukovými simulačními modely v prostředí Macromedia Breeze. Sborník příspěvků MEDSOFT 2005. ISBN 80-86742-07-5, str. 81-88, Zeithamlová M., Agentura Action-M, Praha 2006.
- [5] Stodulka P., Privitzer P., Kofránek J., Mašek J.: Nové postupy v tvorbě simulátorů – inteligentní propojení Matlabu a Simulinku s platformnou .NET a tvorba stavových automatů řídicích výslednou aplikaci. MEDSOFT 2006. ISBN 80-86742-12-1, str. 171-176, Zeithamlová M., Agentura Action-M, Praha 2006.

### Poděkování

*Práce na vývoji lékařských simulátorů je podporována grantem MŠMT č.2C06031 a společností BAJT servis s.r.o.*

Jiří Kofránek  
Laboratoř biokybernetiky  
Ústav patologické fyziologie 1.LF UK  
128 53 Praha 2  
tel: 777-68-68-68  
fax:26731 0503  
e-mail: kofranek@cesnet.cz