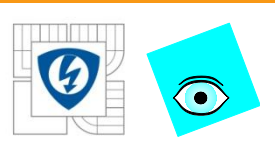


Vizualizace dynamických systémů v prostředí virtuální reality

Jan Daněk, HUMUSOFT s.r.o.
danek@humusoft.cz

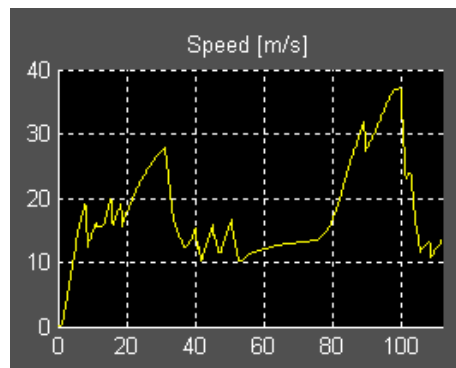
20. ledna 2012

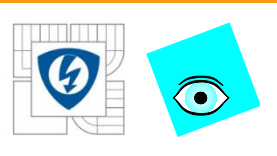
Tato prezentace je spolufinancována Evropským sociálním fondem a státním rozpočtem České republiky.



Vizualizace dynamických systémů

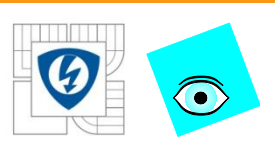
- Inter-disciplinární obor
 - Řídicí technika <> Počítačová grafika
- Grafici:
 - Animace objektů s respektováním jejich skutečné dynamiky
- Technici:
 - Lepší porozumění chování vyvíjených / studovaných systémů
 - Efektivita, rychlost, interaktivita, nová uživatelská rozhraní, cena
- Obrázek je více než 1000 slov... .. a také více, než 10 grafů





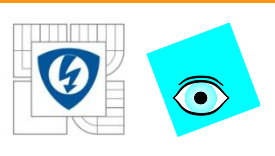
Aplikační oblasti

- Obecné vizualizace
 - Zrychlení vývoje, testování modelu, lepší porozumění
- Modelování interakcí mezi objekty ve virtuální realitě
- Prezentace výsledků simulace v síti WWW
 - Spolupráce mezi vývojovými týmy, Distant Learning
- Virtual prototyping
 - Virtuální laboratorní experimenty
- Ergonomické studie
- Stereovize
- Haptické systémy
- Zpracování videa a obrazu

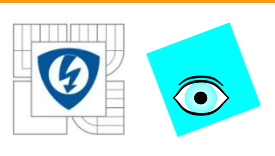


VRML 97

- Hierarchická struktura objektů
- Objekty = Uzly (Nodes)
 - Geometrická primitiva (box, cone...)
 - Obecné tvary (FaceSet, LineSet, ElevationGrid...)
 - Světla
 - Kamery (Viewpoints)
 - Senzory (TouchSensor, ProximitySensor...)
 - Interpolátory (PositionInterpolator...)
 - Skripty, ROUTE, Prototype
 - Materiály, Textury, Video, Audio
- Pole (Fields) – parametry určující vlastnosti objektů



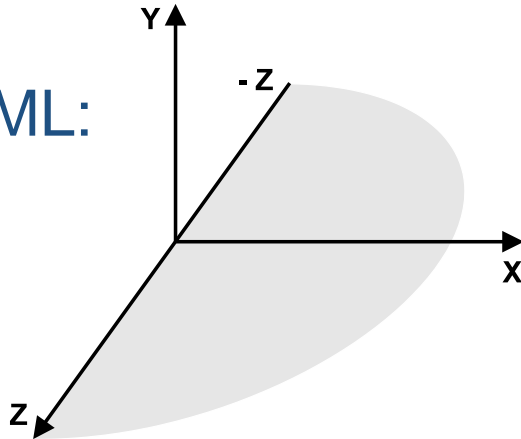
- **External Authoring Interface**
 - **Mechanismus pro přístup k virtuální scéně externími aplikacemi**
 - Simulink 3D Animation používá EAI jen ve vzdálené konfiguraci s použitím WWW browseru – jinak pracuje přímo se scénou
- **Mocný a pružný jazyk pro popis interaktivních 3D scén – ideální pro technické aplikace**



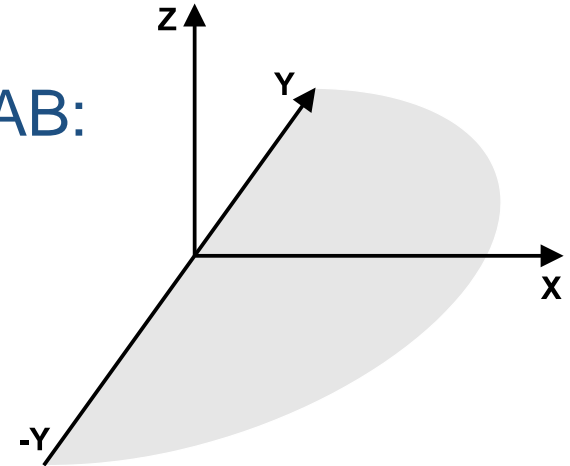
VRML97 – souřadný systém, jednotky

- Osy

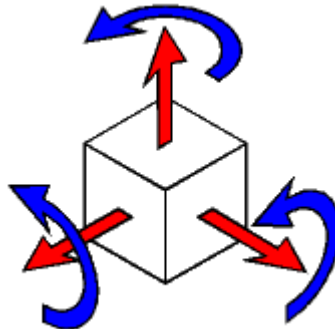
VRML:



MATLAB:



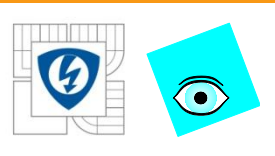
- Rotace



- Jednotky

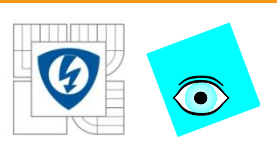
– metry, sekundy, radiány
VRML97

-> norma



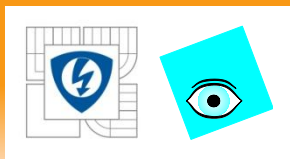
Tvorba virtuálních scén

- Abychom mohli vizualizovat své systémy, musíme si nejdříve připravit příslušnou **virtuální scénu**
- Virtuální scéna obvykle obsahuje:
 - Objekty, které chceme animovat a vizualizovat
 - Okolní prostředí:
 - Pozadí
 - Světla
 - Pohledy,...



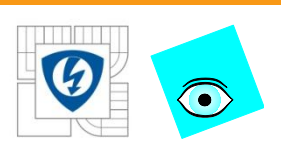
Tvorba virtuálních scén

- Nástroje:
 - VRealm, 3D World Editor
 - Další VRML a 3D editory
 - Import z podnikových CAD systémů
 - Import modelů od poskytovatelů 3D obsahu
 - Manuální tvorba s použitím textového editoru (nejlépe se zvýrazněním syntaxe)
 - **Všechny metody můžeme kombinovat!**



Dobré zvyky při tvorbě VR scén

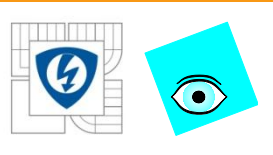
- Vytvořte více pohledů, abyste upoutali pozornost uživatele na to, na co má
- Světla – ve scéně by měly být patrné důležité objekty bez ohledu na „Headlight“, který si může uživatel omylem vypnout
- Definujte pole uzlů WorldInfo a NavigationInfo
 - Velikost scény, viditelnost, „Headlight“, navigační režim „Examine“
- Složené rotace – přizpůsobte se dynamickému modelu
 - Bud' nechte rotace na modelu (s respektováním globálních/lokálních souřadnic)
 - Anebo vytvořte několik vnořených uzlů typu Transform a otáčení rozložte na otáčení každé úrovně podle jedné osy (viz **vrpend**)
- V případě složitých scén můžete použít uzel LOD - Level of Detail
 - Každý objekt má několik grafických reprezentací, objekty vzdálené pozorovateli jsou zjednodušené



Dobré zvyky při tvorbě VR scén

Při importu

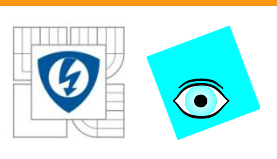
- Objekty a jejich důležité osy zarovnejte s osami souřadného systému
 - Budete s nimi pak moci lépe manipulovat
- Objekty obalte ještě jedním uzlem Transform
 - S největší pravděpodobností budete muset importovaný objekt posunout, otočit a změnit měřítko tak, aby „pasoval“ do vaší scény
- Vyhněte se milionům trojúhelníků (zejména při importu z CAD nástrojů)



DEF

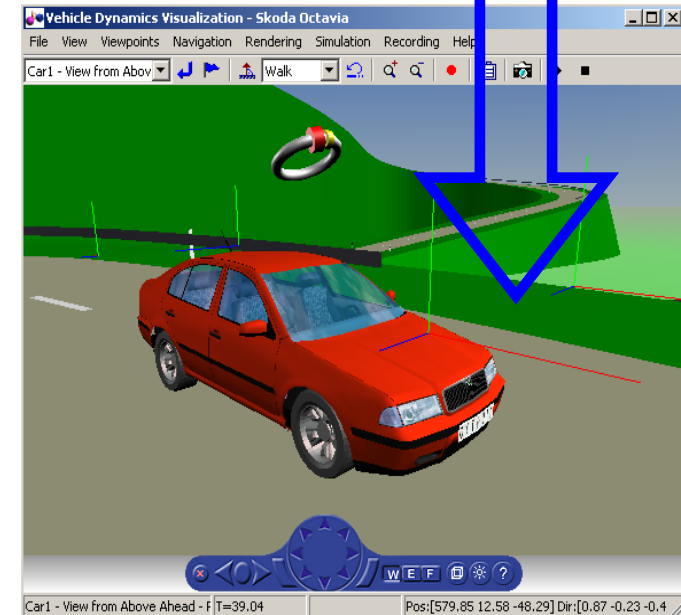
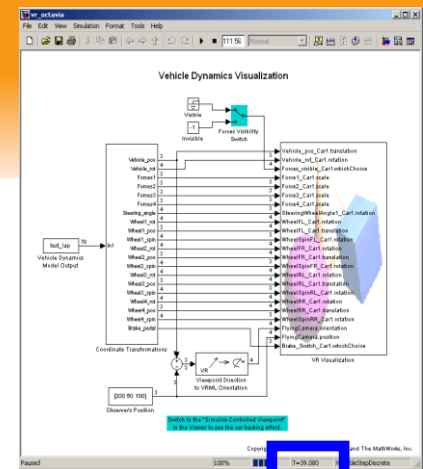
- Pouze ty objekty ve virtuálním světě, které pojmenujete, jsou přístupné prostřednictvím Simulink 3D Animation!
 - Vychází z filozofie EAI
 - Filtrování „Nezajímavých“ objektů
- VRML97 – příkaz DEF

```
DEF Auto Transform {  
  ...  
}
```
- Jména můžou objektům přiřazena buď s použitím jakéhokoliv VRML editoru, nebo také textového editoru



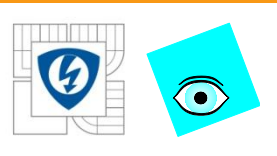
Simulink 3D Animation

- Propojuje MATLAB® a Simulink® s virtuální realitou
- Prohlížení a interakce s 3D reprezentací simulačních modelů
- Umožňuje MATLABu a Simulinku řídit polohu, rotaci a velikost objektů definovaných ve virtuálním světě
- Signály ze senzorů ve virtuálním světě mohou být čteny a zpracovávány simulačním modelem
- Vizualizace v reálném čase
- Podpora vstupních zařízení
 - Joystick, SpaceMouse
- Součástí programového balíku MATLAB



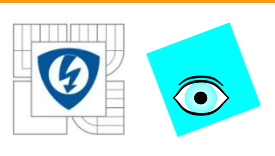
Příklad:

Vizualizace HIL experimentu (testování řídicí jednotky ESP - Škoda Octavia)



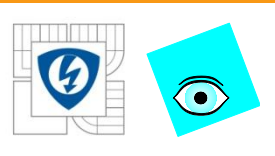
Simulink 3D Animation

- **Otevřená filozofie**
 - **Založen na jazyce VRML97**
 - VRML = Virtual Reality Modeling Language
 - International Standard ISO/IEC 14772-1:1997
 - Otevřený standard pro popis virtuálních scén
 - Podporovaný mnoha 3D a CAD systémy
- **Architektura Client / Server**
 - **Server – hostuje simulační model**
 - MATLAB, Simulink
 - **Client – pozorování animovaného modelu ve VR, interakce v rámci virtuální scény**
 - Client může pracovat na tomtéž počítači jako Server

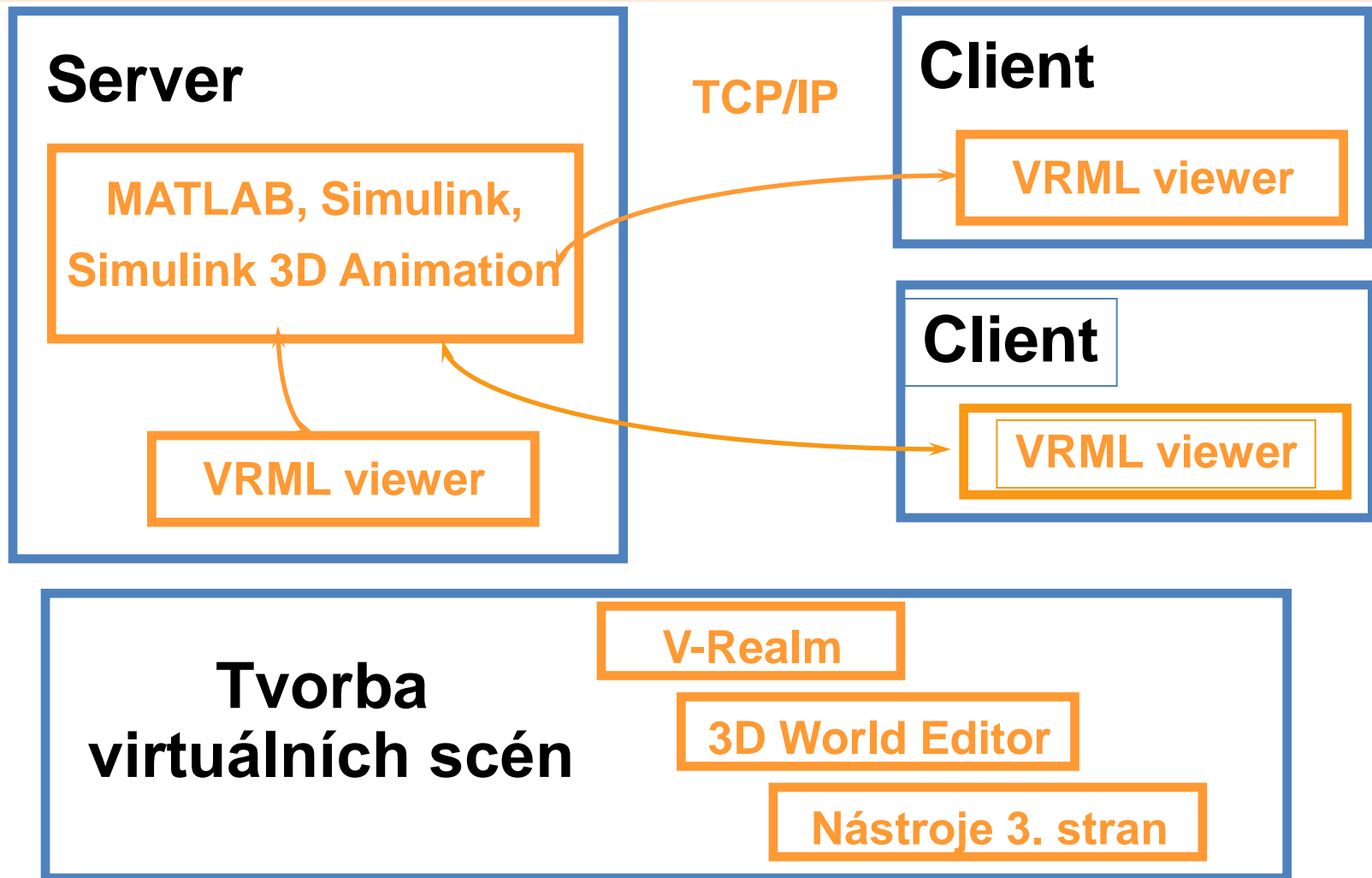


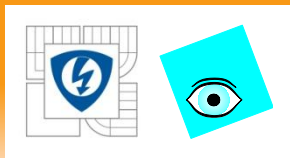
Simulink 3D Animation

- Kompletní vývojové prostředí
 - Prohlížeče (Interní, Orbisnap, blaxxun Contact)
 - Editor (V-Realm, 3D World Editor)
 - Nástroje pro analýzu experimentů (vrplay)

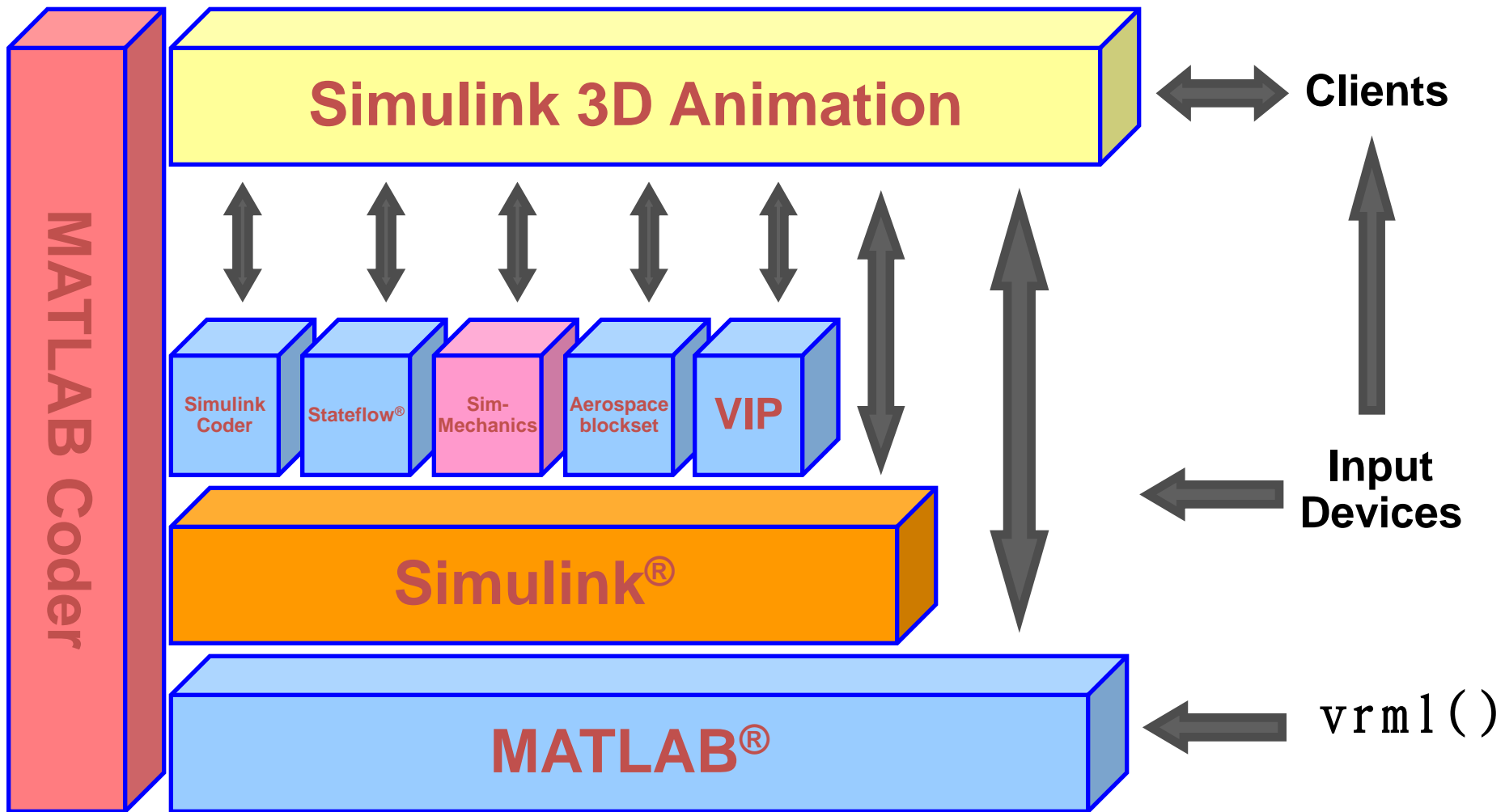


Simulink 3D Animation - Architektura





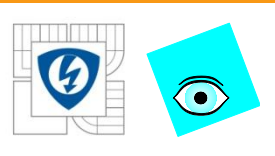
Simulink 3D Animation v rámci systému MATLAB



20.1.2012

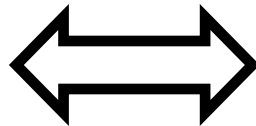
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ





Rozhraní MATLAB vs. Simulink

- objekt **vrworld** je sdílen oběma rozhraními
- Simulink interface – uživatelsky příjemné a snadné
 - Dynamické systémy, prototyping, interakce
 - Snadné nahrávání offline animací
- MATLAB interface – zároveň infrastruktura pro Simulink Interface i samostatně využitelné pracovní prostředí
 - Všeobecné výpočty, FEM
 - Cokoliv, kde nezávislá proměnná může být interpretována jako čas
 - Jednorázové operace typu View | Get | Set | **Save**

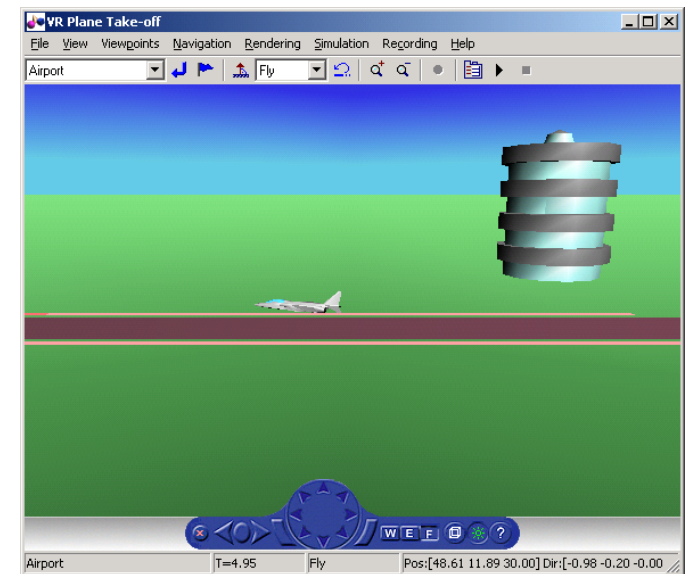
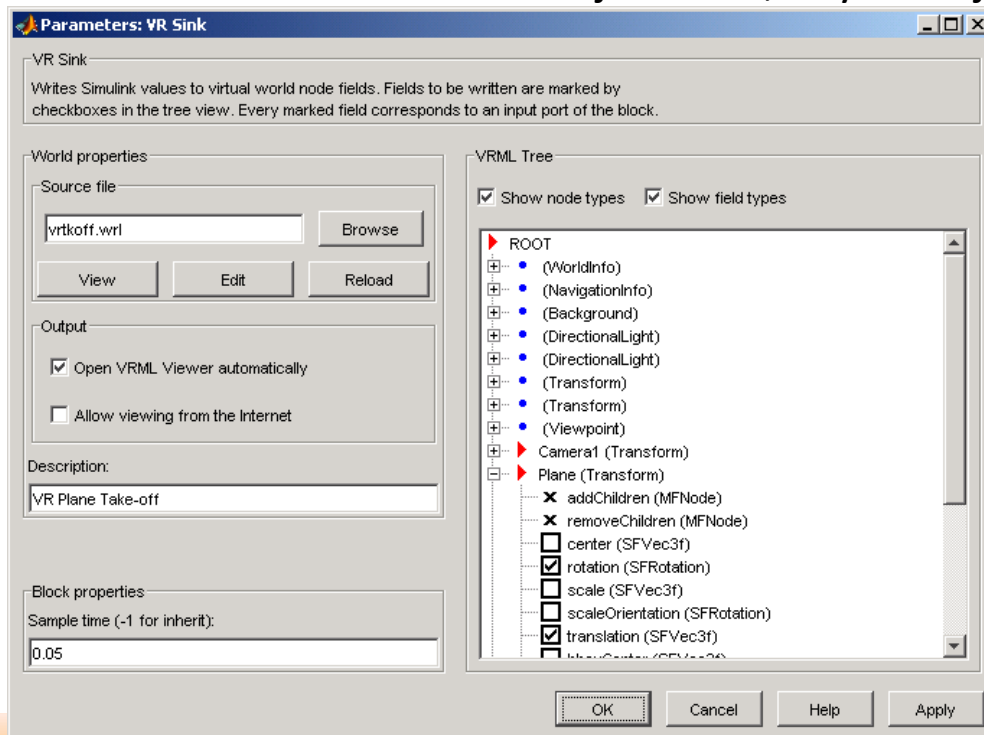


```
wh=vrworld('vr_fem.wrl');  
open(wh);  
nh.FEM_Colormap=vrnode(wh,'FEM_Colormap');  
nh.FEM_Coords=vrnode(wh,'FEM_Coords');  
nh.FEM_IFS=vrnode(wh,'FEM_IFS');  
set(nh.FEM_Coords,'point',v);  
set(nh.FEM_IFS,'coordIndex',e);  
set(nh.FEM_IFS,'colorIndex',ci);  
set(nh.FEM_Colormap,'color',cm);  
view(wh);
```

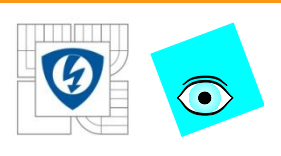


Práce s Simulink 3D Animation

- Asociujte Simulinkový model s virtuální scénou
- Signálům Simulinku přiřadte vlastnosti objektů ve virtuálním světě
- Vytvořte vhodné pohledy, aby bylo možné sledovat pohyb objektů
- Během simulace můžete změnou hodnoty signálů manipulovat s polohou, otočením a dalšími vlastnostmi 3D objektů tak, aby co nejvěrněji popisovaly reálný objekt



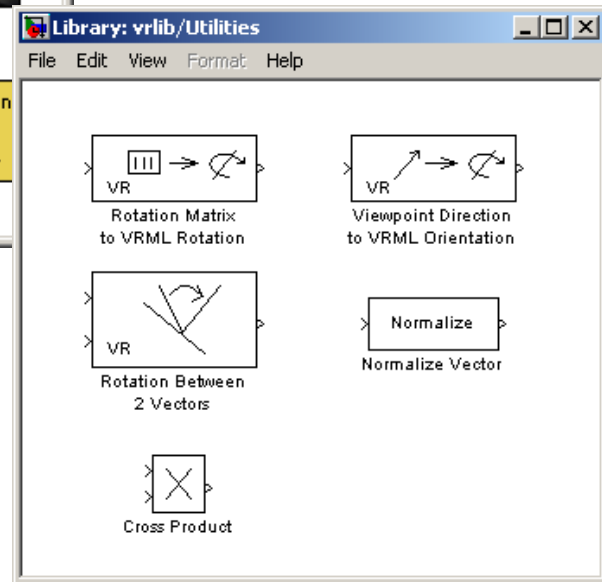
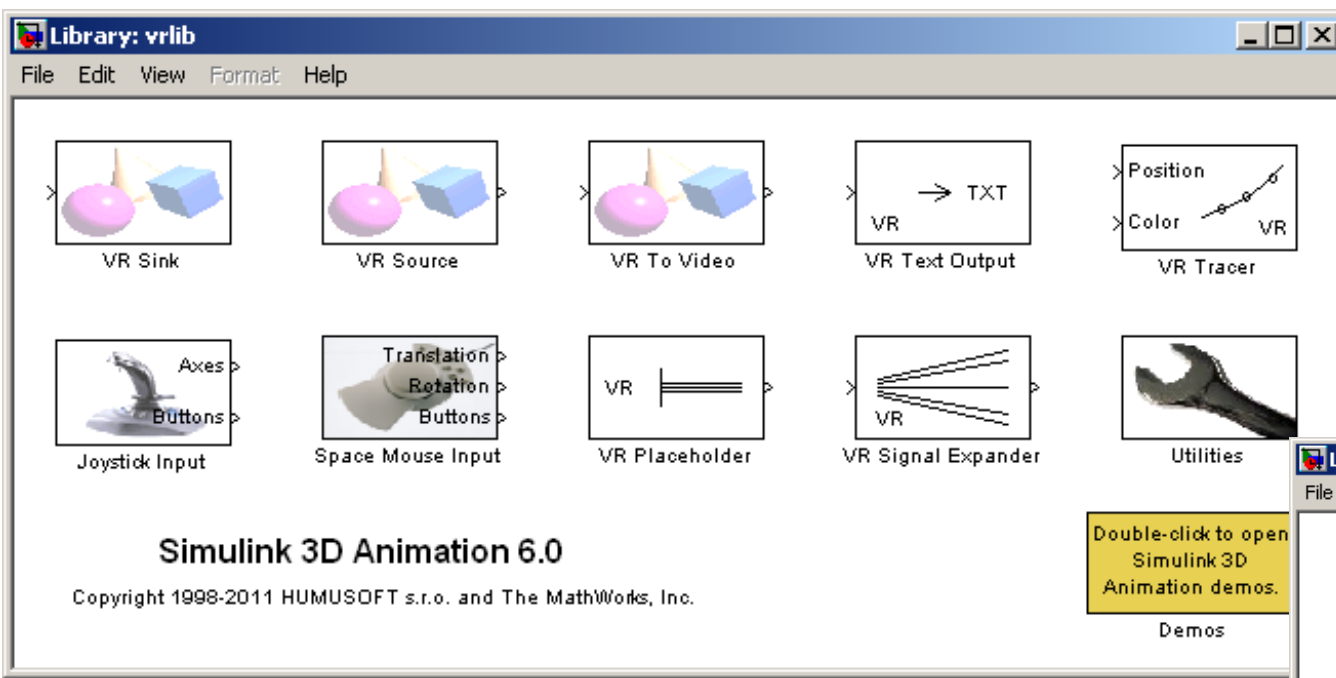
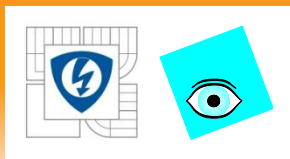
20.1.2012



Prohlížení simulace ve Virtuální Realitě

- **Simulink 3D Animation viewer**
 - Základní nástroj pro prohlížení scén
 - Integrovaný se Simulinkem (Start / Stop atd.)
- **vr.canvas**
 - Začlenění 3D obsahu do okna MATLAB Figure (GUI apod.)
- **VRPLAY**
 - Sledování předem nahraných offline VRML animací
- **Web browser**
 - VRML plug-in (blaxxun Contact)
 - Zvuk, kompatibilita
- **Orbisnap (www.orbisnap.com)**
 - Samostatná, volně šiřitelná aplikace
 - Sledování předem nahraných offline VRML animací
 - Rychlé prohlížení VRML modelů
 - Vzdálené připojení k světům obsluhovaným SL3D Serverem

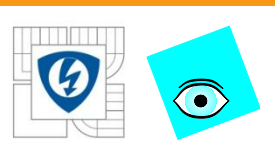
Simulink 3D Animation – Knihovna bloků



20.1.2012

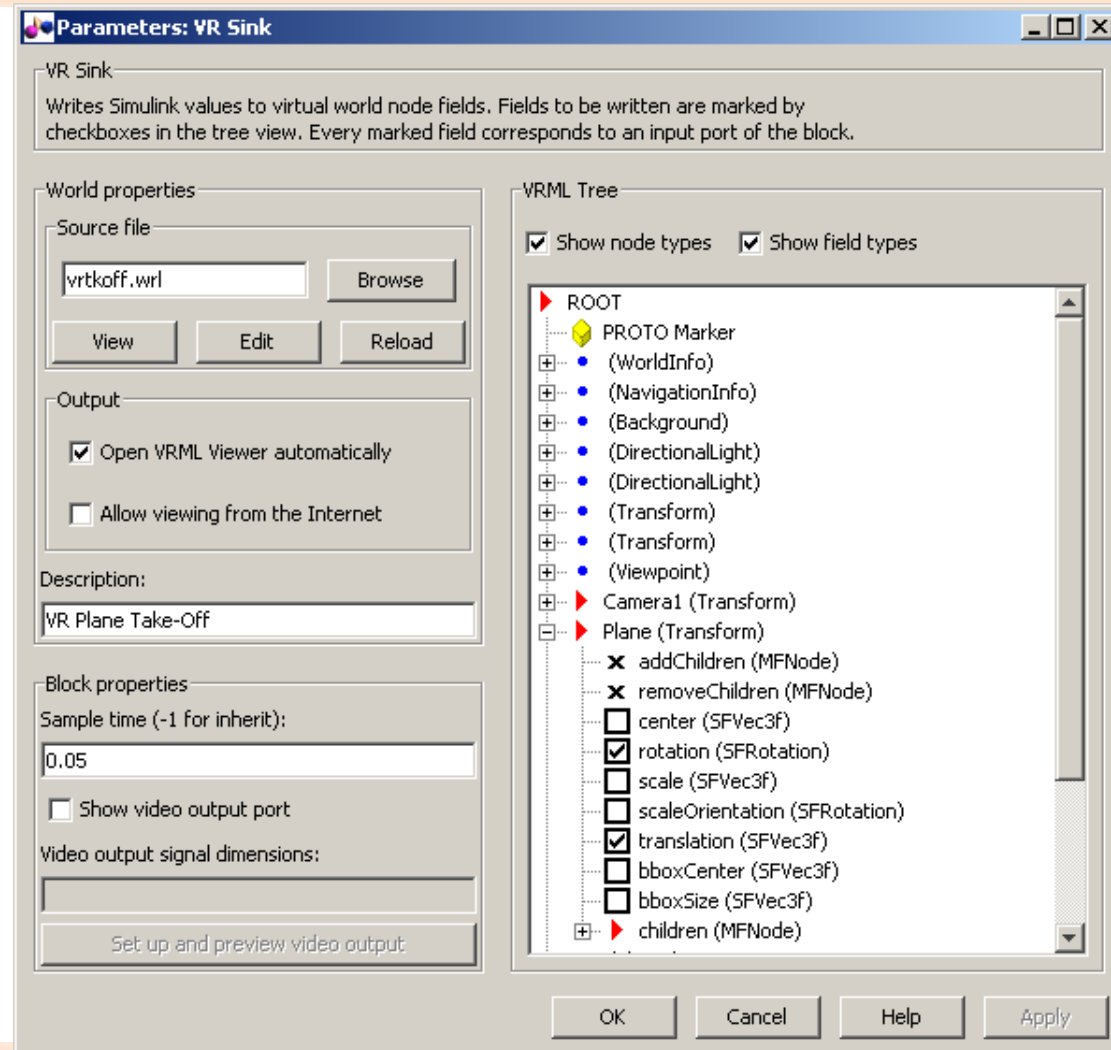
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



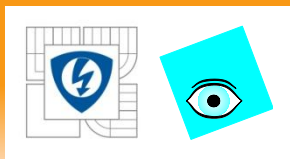


VR Sink / VR To Video / VR Source

- Základní bloky pro asociaci modelu s VR scénou
- VR Sink / VR To Video
 - Výstup signálů do VR
- VR Source
 - Čtení signálů z VR



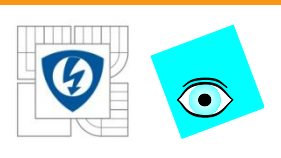
20.1.2012



Joystick Input, SpaceMouse Input

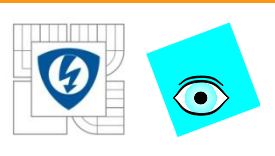
- **Asynchronní** vstupní zařízení, použitelná i mimo práci s virtuální realitou
- Používají Windows **HID (Human Interface Device)** interface
 - automatická konfigurace právě připojených zařízení
 - je potřebné s tímto počítat při tvorbě modelů
(co se stane, když Joystick jiného uživatele modelu bude mít jiný počet aktivních os než můj Joystick?)
- Joystick Input - Výstupy: **Positions, Rotations, Buttons, POV**
- **SpaceMouse Input** (pracuje se všemi zařízeními společnosti 3Dconnexion)
 - 3 režimy práce:
 - speeds
 - positions
 - viewpoint control

-> vrccrane_joystick, vrmanipul



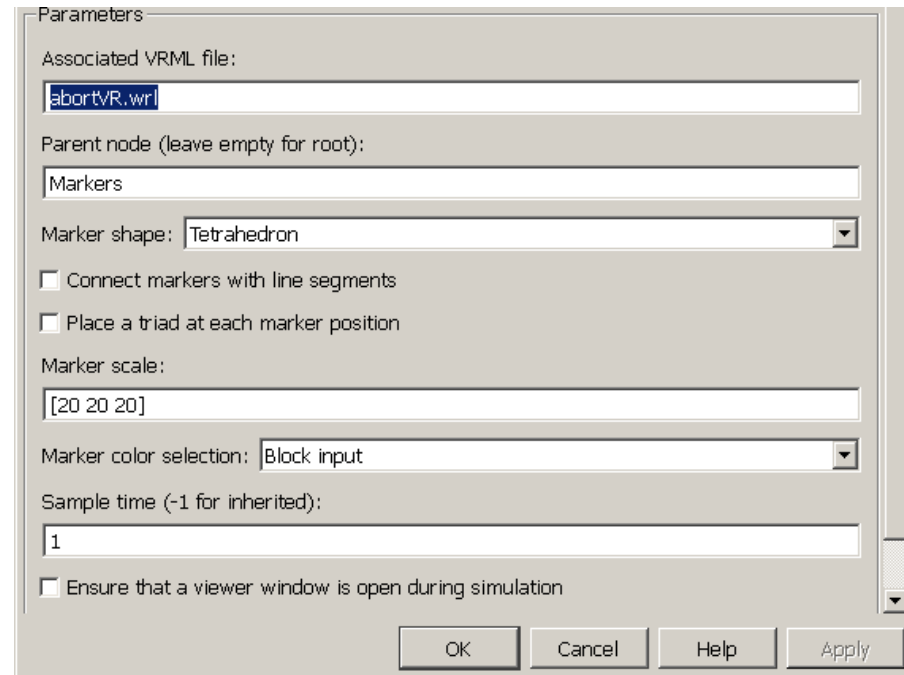
VR Signal Expander, VR Placeholder

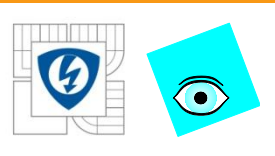
- Podpůrné bloky
- VR Placeholder signál = „Neměň danou hodnotu“
 - posláním vektoru **[VRP 8 VRP]** do pole představujícího polohu objektu nastavuji jeho Y souřadnici na 8, ostatní souřadnice objektu zůstávají beze změny
- VR Signal Expander
 - Sestavování a přeskupování vektorových signálů tak, aby odpovídali struktuře polí VRML.
 - Mapování vstupního vektoru na výstupní, chybějící pozice ve výstupním vektoru doplněny o signál VRP
 - Snadná záměna souřadnic - $[x \ y \ z] \rightarrow [x \ z \ y], \dots$



VR Tracer

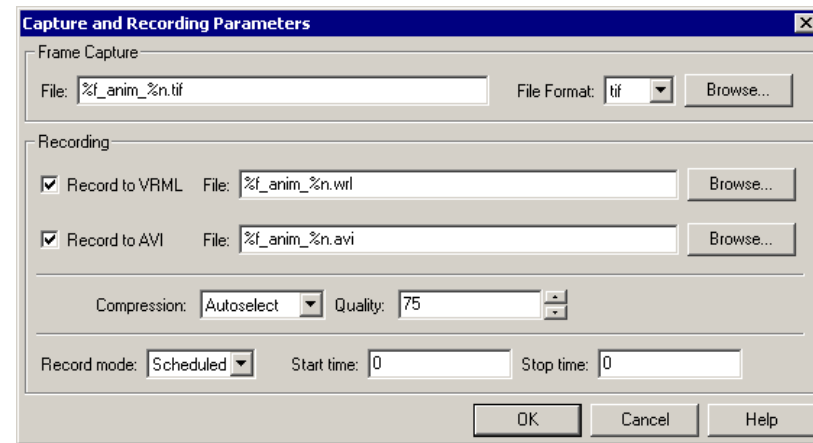
- Sledování trajektorie objektů
- Barva trajektorie jako volitelný vstup
- R2012a:
 - Průmět trajektorie do roviny
 - Spojnice trajektorie s bodem
- -> vrtkoff_traj
- -> sf_launchabort

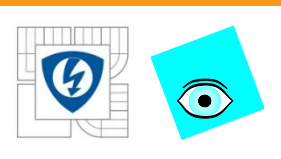




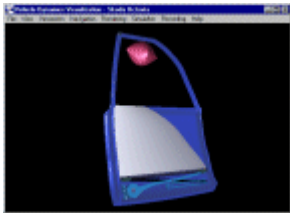
Záznam animačních souborů

- Možnost záznamu animačních souborů pro pozdější přehrávání
- Soubory mohou být přehrávány nezávisle na prostředí MATLAB/Simulink
- Distribuce, dokumentace a publikace výsledků simulace
- 3D – Animační soubory VRML
 - Malá velikost
 - Stejná kvalita grafiky jako původní scéna
 - Navigace ve virtuální scéně
 - Vyžaduje VRML prohlížeč (Orbisnap)
- 2D – Animační soubory AVI
 - Kompromis mezi kvalitou a velikostí souboru
 - Animace uložena přesně tak, jak ji autor zaznamenal během simulace
 - Přenositelnost – většina počítačů má Windows Media Player
 - Dovoluje zaznamenat změny i distribuovaných parametrů, které VRML interpolátory nemohou postihnout (barva povrchu, tvar apod.)





Příklady



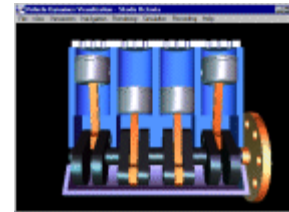
Power Window Controller



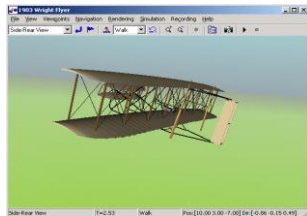
Vehicle Dynamics



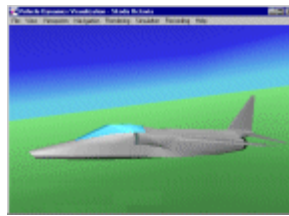
Crash Test with Airbag



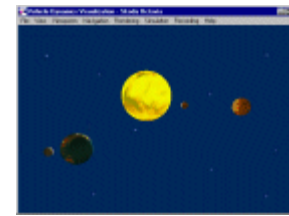
Four Cylinder IC Engine



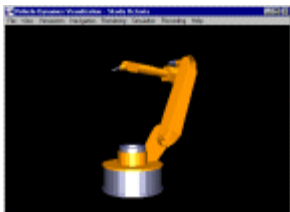
Wright Flyer



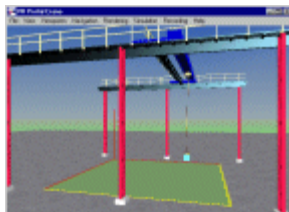
Plane Take-off



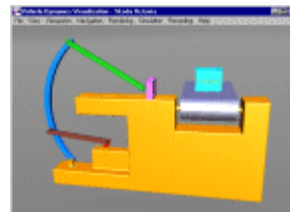
Solar System



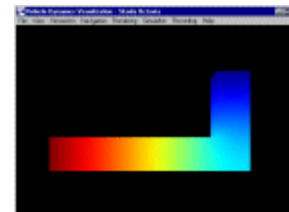
Robot Arm



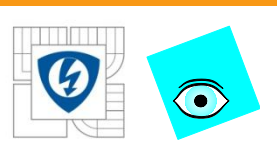
Portal Crane



Conveyor Model



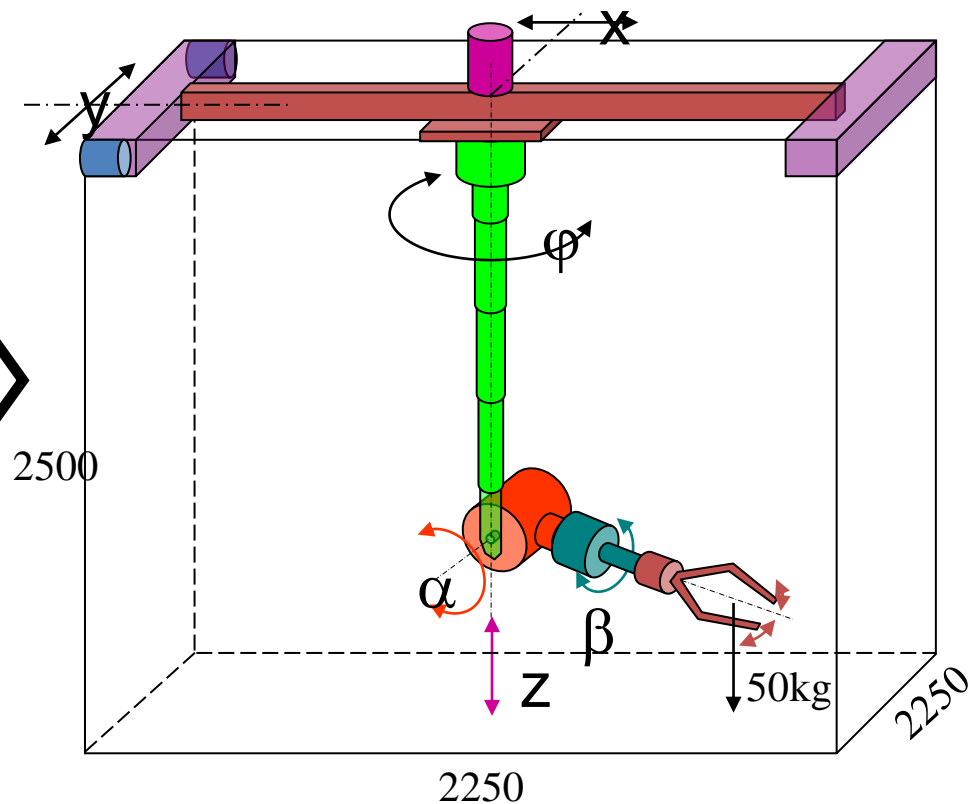
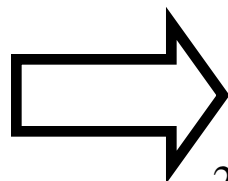
Heat Transfer



VR Prototyping – Control of a Hot chamber Manipulator using SpaceMouse

- Design of a suitable and ergonomically feasible control elements

vrmanipul

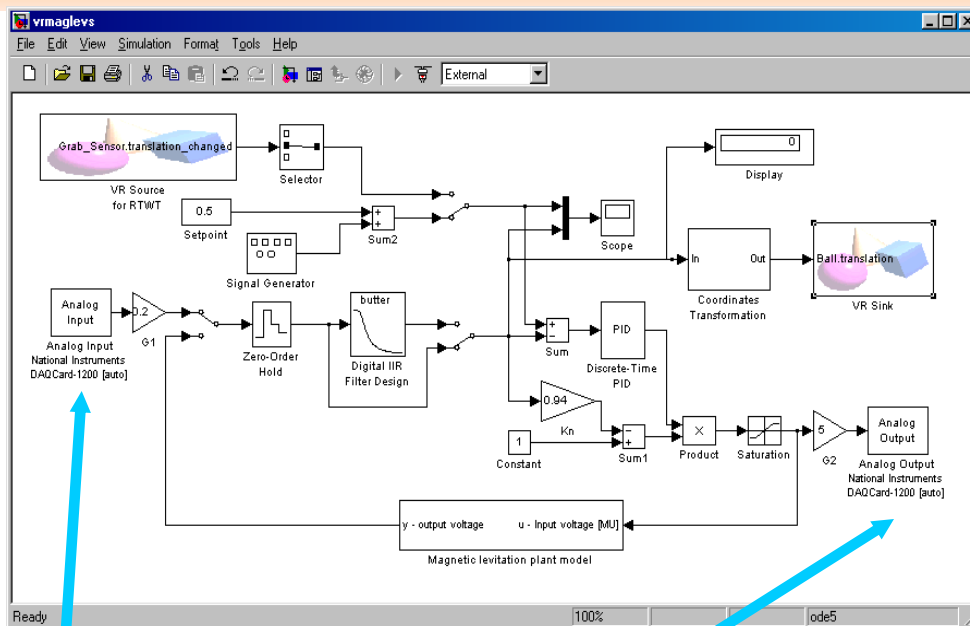


Model courtesy of ProTyS spol. s r.o., Czech Republic

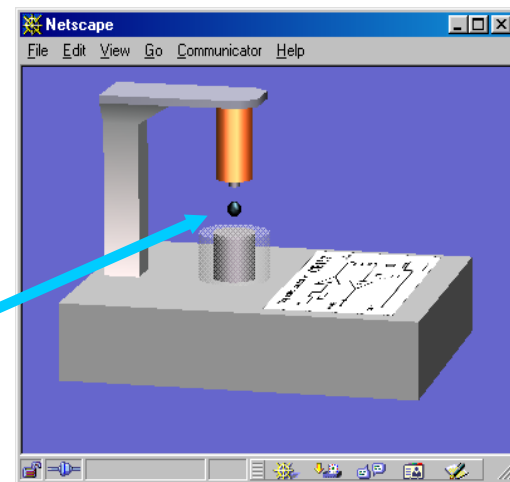
20.1.2012

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Real-Time: Magnetic Levitation Model



- Simulink diagram contains Analog Input / Output blocks.
- Model can be compiled and then run in the **real time** using the "Real Time Windows Target".
- During the runtime, the model interacts with both the hardware model AND the associated VR world.

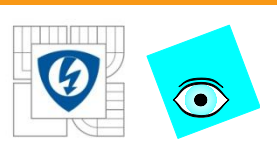


Ball:
VRML plane
sensor

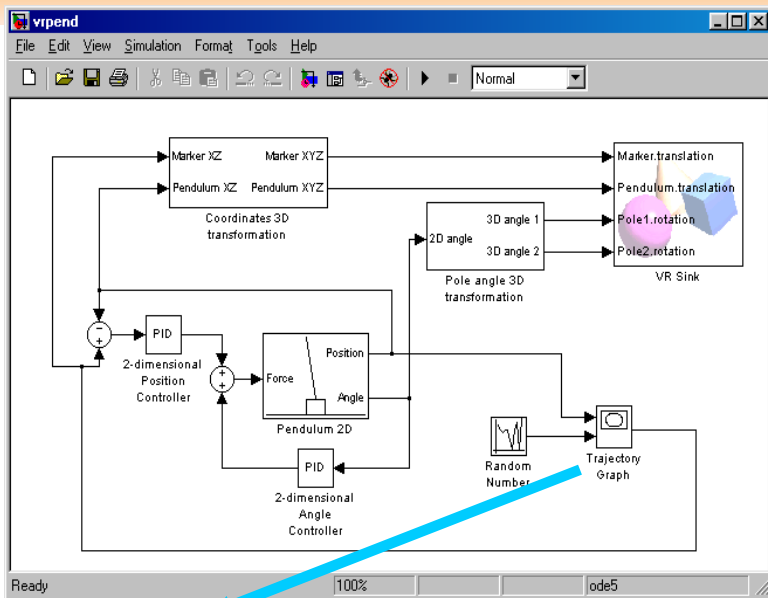
Analog Output:
Magnetic coil
current

Analog Input:
Inductive sensor = ball position

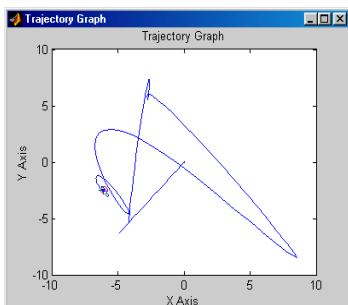
`vrماغlev, vrماغlev_rtw`



VRML Sensors: Inverted Pendulum

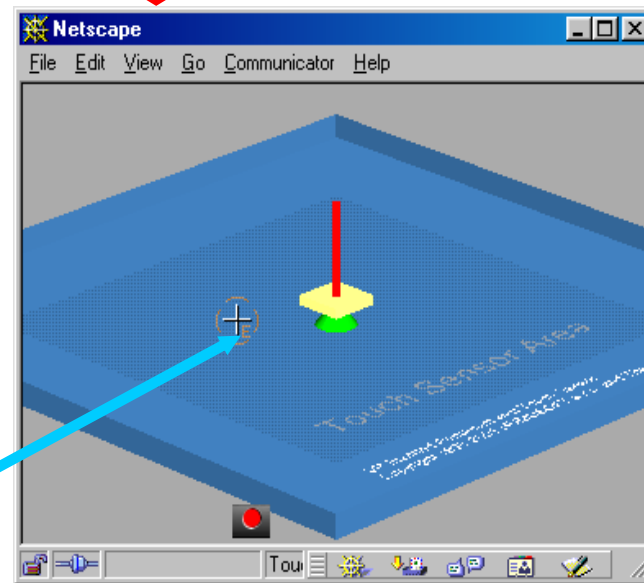


Setpoint translation,
pendulum translation
and rotation controlled
from simulation model

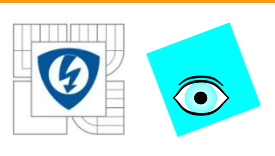


Trajectory Graph:
Position trace

Touch Sensor:
Changing the setpoint

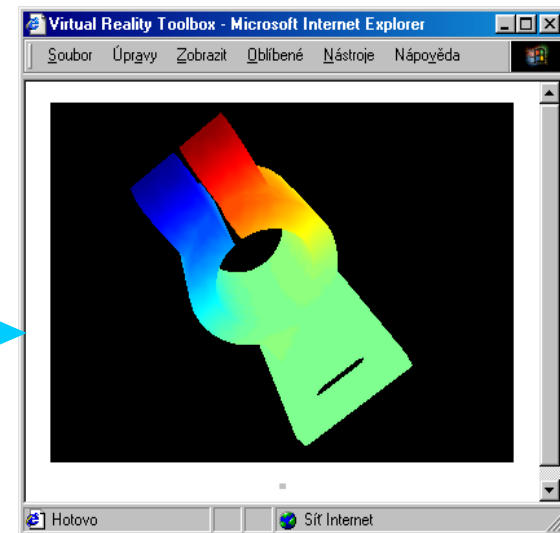
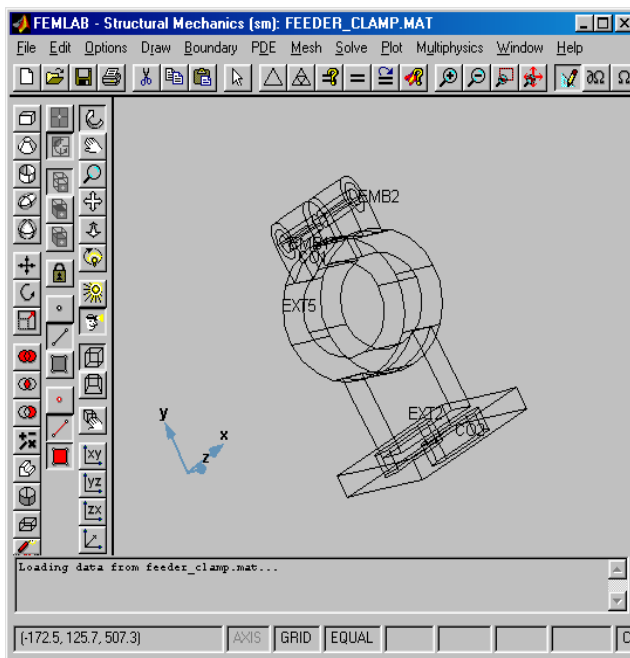


vrpend



Multiphysics in MATLAB: FEM and VRML

- COMSOL Multiphysics - application for FEM analysis
- Simulink 3D Animation:
 - Dynamic export of arbitrary data structures into VRML (also animation, including MF fields...)
 - Mapping of values to colours

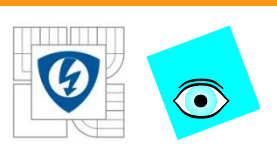


**Static deformation of a clamp
when a load is applied on a feeder
fastened in the clamp**

vrheat

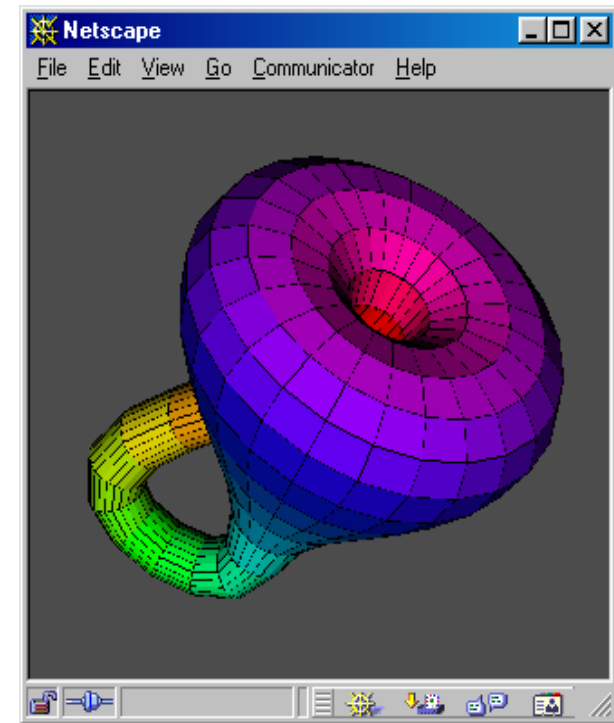
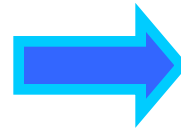
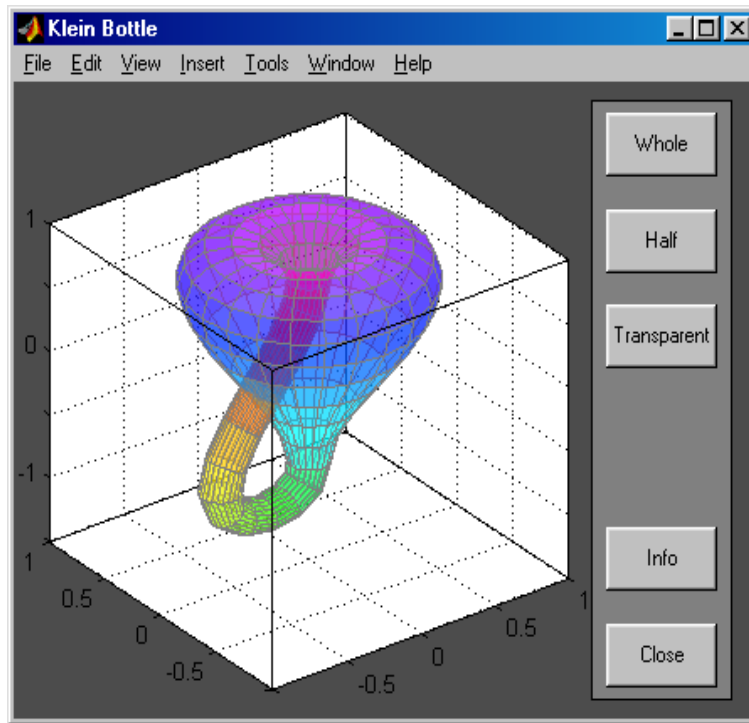
20.1.2012

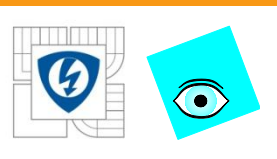
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



MATLAB a VRML „Před Simulink 3D Animation“

- `vrml(object_handle, 'file.wrl');`
 - standardní funkce pro statický export objektů Handle Graphics objects do formátu VRML97

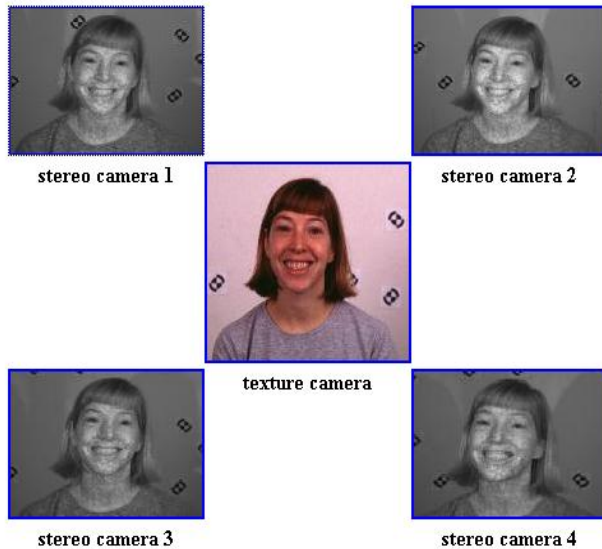




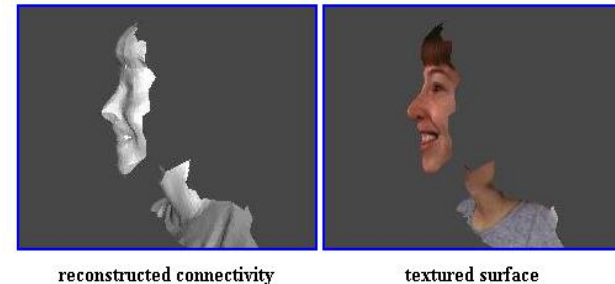
MATLAB a VRML „Před Simulink 3D Animation“

3D Model Reconstruction from Stereo Vision

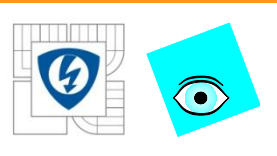
Input data from calibrated cameras



Surface reconstructed by linking local models

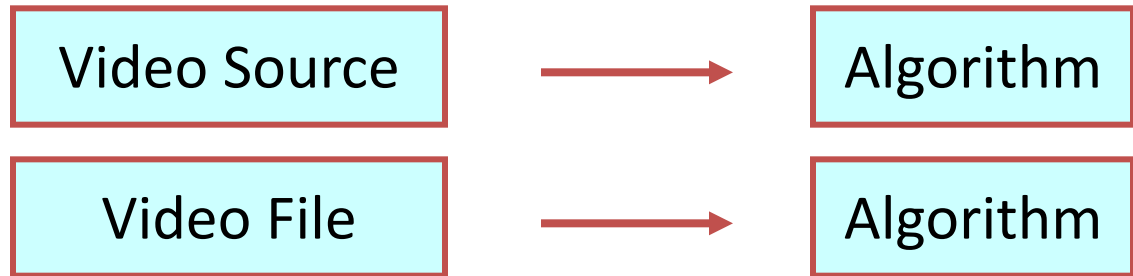


Courtesy of Center for Machine Perception, Czech Technical University, Praha (<http://cmp.felk.cvut.cz>)

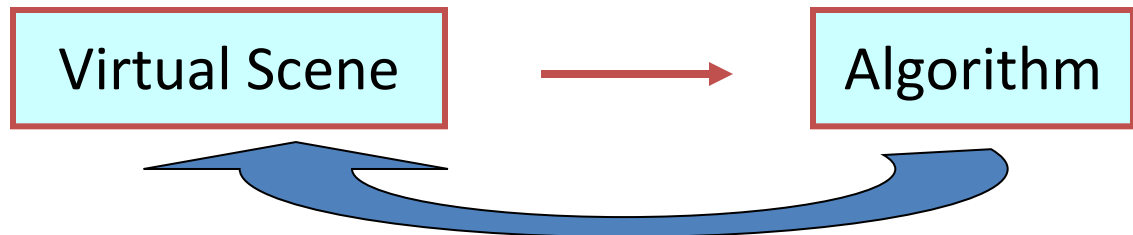


Zpracování Videosignálu

- **Typický scénář:**



- **Scénář s použitím VR:**



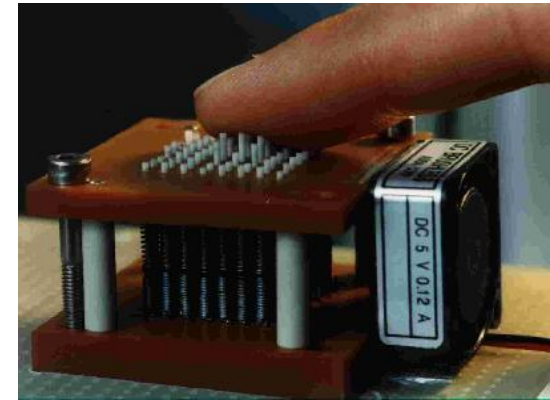
- **Výhody:**

- Možnost zpětné vazby z výstupu VIP algoritmu zpět na vstup, který ovlivňuje objekty ve virtuálním světě
- -> Uzavření regulační smyčky!
 - V případě zpracování video souborů nemožné
- Řádově menší soubory potřebné k testování algoritmů



Haptické aplikace

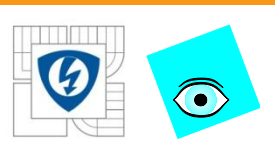
- Human Haptics
 - Mechanika a psychologie lidského hmatu
- Machine Haptics
 - Vývoj zařízení pro zprostředkování hmatu
- Computer Haptics
 - Algoritmy a metody umožňující zprostředkování hmatových vjemů, implementované pomocí výpočetní techniky
 - Analogie pojmu počítačová grafika
 - Analogicky k počítačové grafice, haptická zařízení jsou často nazývána *haptické displeje*



Taktilní displej

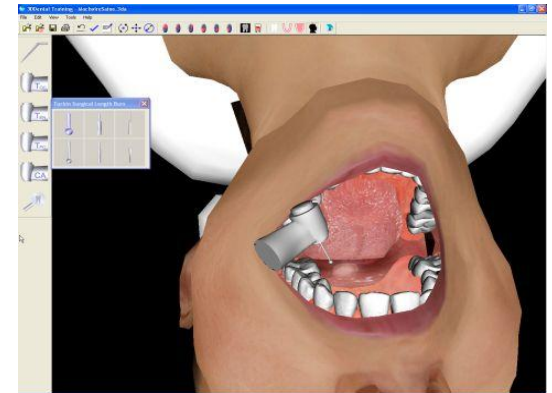
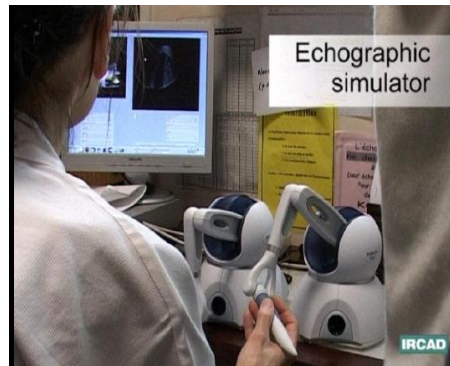


SensAble PHANTOM



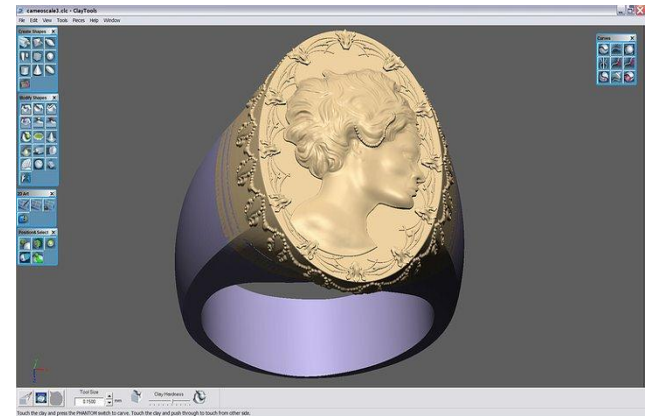
Příklady haptických aplikací

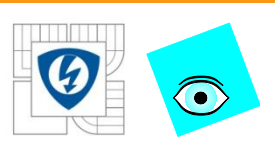
- Medicína - trenažery práce s ultrazvukovým diagnostickým přístrojem (sonografem), dentistickými přístroji apod.



Produktivita CAD návrhů

ClayTools™ umožní designérům použít hmatové smysly pro tvorbu organických tvarů a přidávat do svých NURBS modelů v CAD aplikacích sochařské detaily, ruční úpravy, kompletní přechody tvarů a reliéfní textury (žlutá barva reprezentuje virtuální hlinu, fialová NURBS). Implementováno pro Rhino, 3DS Max, Maya.

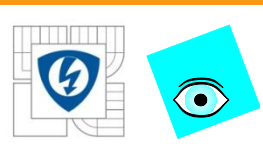




Haptický systém - komponenty

< Telehaptické systémy >

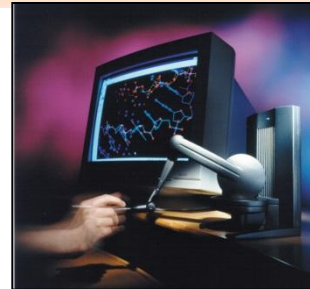
- Haptická zařízení
 - Zařízení ovládaná silou, poskytují zpětnou vazbu až v 6 stupních volnosti
 - Uživatel je používá k interakci s virtuálním / vzdáleným prostředím
- Systém zprostředkující vizuální zpětnou vazbu od reálného nebo virtuálního prostředí - Visual Rendering
 - Vizuální prostředí, se kterým uživatel interaguje může být buď reálné (kamery) nebo syntetické (VR modely) (~30 Hz)
- Systém zprostředkující haptickou zpětnou vazbu - Haptic Rendering
 - Regulátor + haptické efekty + synchronizace haptických událostí s prostředím
 - Rychlost zpětné vazby mnohem vyšší než u vizuální zpětné vazby (~1 kHz)
- Síťová infrastruktura
 - Telehaptické systémy = několik lokálních haptických systémů propojených v síti, umožňujících interakci na dálku
- Systém pro kompenzaci zpoždění v síti
 - Telehaptické systémy jsou velmi citlivé na zpoždění v regulační smyčce
 - 50ms již může způsobit nestabilitu systému - nutnost aktivní kompenzace zpoždění



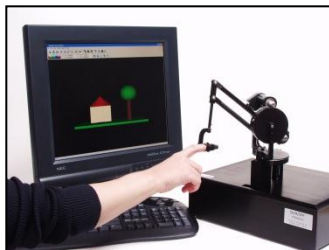
Haptická zařízení



PHANTOM® Omni™



PHANTOM® Desktop™



Premium 1.0



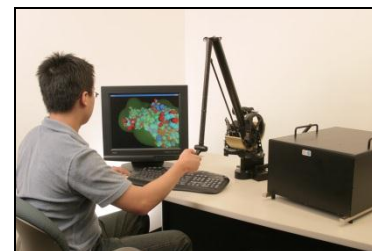
Premium 1.5



Premium 3.0



Premium 1.5/6DOF

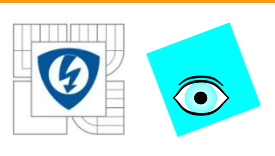


Premium 3.0/6DOF

20.1.2012

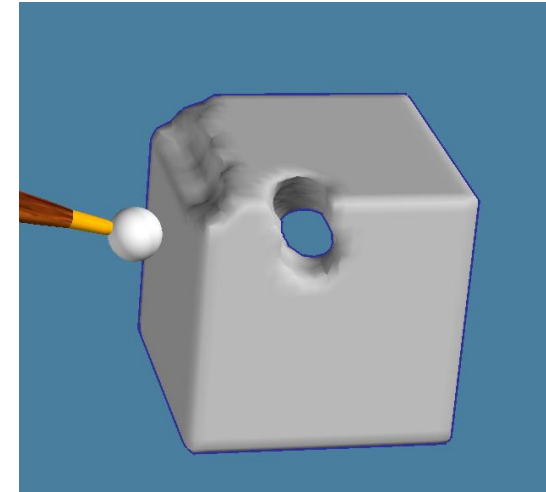
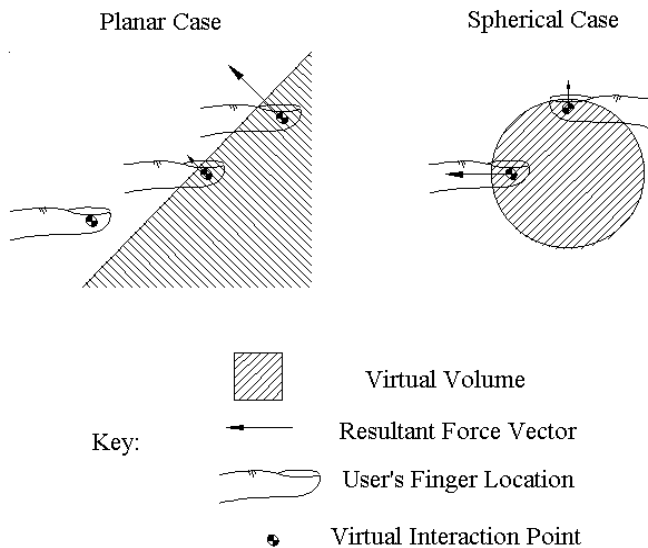
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ





Haptické efekty

Jednoduchý model aplikace síly – Hookův zákon

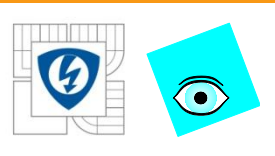


Kolize tvarově složitých objektů

Vyvolaná síla je proporciální hloubce vniknutí do virtuálního objektu

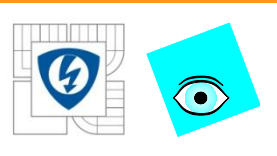
$$F = -kx$$

Aplikace síly na základě výsledků simulace - *haptic rendering*



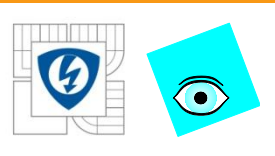
Příklad 1 – MATLAB Interface

```
% Move an object in the virtual world
% Associate scene with a vrworld object, open a view it
wh = vrworld('vrmount.wrl');
open(wh);
view(wh);
% Get list of nodes in the virtual world
nodes(wh);
% 3 ways to inspect Car position
pos = wh.Automobile.translation      nh = wh.Automobile;
                                     pos = nh.translation      pos =
getfield(nh, 'translation')
>> 3.0000      0.2500      0
% Set car position
nh.translation = [3 0.25 10];
wh.Automobile.translation = [3 0.25 10];
setfield(nh, 'translation', [3 0.25 10]);
% Close and delete the vrworld object
close(wh); delete(wh);
```

Příklad 2 – MATLAB Interface

```
%% Read in a loop value of virtual scene object  
fields  
% Read and open world file, open figure with VR  
canvas  
% and set the viewpoint  
w = vrworld('portal_crane');  
open(w);  
c = vr.canvas(w);  
w.Console_Switch.whichChoice = 0;  
c.Viewpoint = 'Console_Closeup';  
%% Loop  
while ~(w.PB_Start.Button_State)  
if (w.PB_XMinus.Button_State) disp ('-X Button  
pressed'); end  
vrdrawnow;  
end
```

Příklad 3 – Simulink Interface

- Otevřete model Simulinku **vrtut2**
- Vložte do modelu blok **VR Sink** z knihovny **vrlib**
- V dialogu bloku VR Sink vyberte veličiny, které chcete řídit z modelu:
 - Plane.rotation
 - Plane.translation
- Vzniklé vstupy bloku VR Sink napojte na příslušné signály dynamického modelu
- Otevřete okno VR prohlížeče
- Spustěte simulaci