



SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

Fakulta informatiky a informačných technológií

Ilkovičova 3, 842 16 Bratislava 4



Robocup 3D

(Tímový projekt)

Používateľská príručka



tím č. 11

HVIEZDNA JEDENÁSTKA

Odbor: Softvérové inžinierstvo, Informačné systémy
Vedúci tímu: Ing. Ivan Kapustík
Autori: Bc. Rastislav Barlík
Bc. Marian Buchta
Bc. Štefan Dlugolinský
Bc. Michal Kvetan
Bc. Stanislava Leitmanová
Bc. Milan Šillík
Dátum: 20. máj 2008

Obsah

1. INŠTALÁCIA PROSTREDIA.....	3
1.1 Inštalácia a nastavenie VMware Workstation vo Windows XP.....	3
1.2 Inštalácia rcssservera 3D v Ubuntu 8.04 Hardy Heron.....	4
1.3 Inštalácia rcssservera 3D pod Mac OS X	6
2. HRÁČ SIRIUS.....	9
2.1 Stiahnutie hráča a adresárová štruktúra projektu	9
2.2 Kompilácia a spustenie hráča	10

1. Inštalácia prostredia

V tejto časti príručky sú uvedené všetky postupy, ktoré súvisia s inštaláciou vývojového prostredia a servera rcssserver3D.

1.1 Inštalácia a nastavenie VMware Workstation vo Windows XP

Najprv je potrebné nainštalovať VMware Workstation (minimálne verzia 6.0.2). Následne vytvoríme virtuálnu mašinu pre Ubuntu. Postup je nasledovný:

1. New virtual machine
2. Virtual machine configuration: Typical
3. Guest operating system: Linux -> Ubuntu
4. Vybrať názov (napr. Ubuntu 7.10, 8.04) a umiestnenie, kde sa budú nachádzať súbory (rezervovať cca. 4GB)
5. Network connection: NAT (alebo podľa potreby)
6. Disk capacity (nechať tak ako je, pretože sa disk neskôr nahradí novým)
7. Finish

Po vytvorení virtuálnej mašiny sú potrebné menšie úpravy. Vo VMware Workstation zvolíme Edit virtual machine settings:

- zvolíte toľko RAMky koľko potrebujete, stačí 512MB), kliknúť na floppy a vybrať Remove
- Ak potrebujete 2 jadrá, zvolíte Processors na hodnotu 2

Treba nastaviť frekvenciu procesora, pretože málokedy uhádne správnu. Zistíte si prednú frekvenciu vášho procesora. Bude sa zadávať v kHz (napr. ja mám 1730000 kHz) (vypnite si VMware program, pretože sa vám môže stať, že sa vám opäť vymaže súbor config.ini)

Pre XPčka:

- Choďte do
`c:\Documents and Settings\All Users\Application Data\VMware\VMware Workstation\config.ini` .
 Ak súbor nenájdete, vytvorte ho.
- Napíšte do neho toto:

```
host.cpukHz = "1730000"
host.noTSC = "TRUE"
ptsc.noTSC = "TRUE"
```

Po spustení virtuálnej mašiny je vhodné stlačiť klávesu F2, čím sa dostaneme do BIOSu. V nej treba odstrániť Floppy mechaniku.

Návod na inštaláciu VMware Tools v Ubuntu 8.04

1. Spustiť ubuntu.
2. Vo virtuálnej mašine zvoliť VM -> Install VMware Tools... (na ploche sa zobrazí mechanika)
3. Spustiť terminál a zadať príkazy:
 - `sudo apt-get update`
 - `sudo apt-get install build-essential libgtk2.0-dev`
 - `sudo apt-get install libproc-dev libdumbnet-dev xorg-dev`
 - `cd Desktop/`

- `wget http://mesh.dl.sourceforge.net/sourceforge/open-vm-tools/open-vm-tools-2008.04.14-87182.tar.gz`
- `cp -a /media/cdrom/VMwareTools* .` (nezabudnúť na bodku)
- `tar xzvf VMware*.gz`
- `tar xzvf open-vm-tools*.gz`
- `cd open-vm-tools-2008.04.14-87182/`
- `./configure && make`
- `cd modules/linux/`
- `for i in *; do mv ${i} ${i}-only; tar -cf ${i}.tar ${i}-only; done`
- `cd ../../..`
- `mv -f open-vm-tools-2008.04.14-87182/modules/linux/*.tar vmware-tools-distrib/lib/modules/source/`
- `cd vmware-tools-distrib/`
- `sudo ./vmware-install.pl`
- Ak sa počas inštalácie bude na niečo pýtať, stlačiť Enter. Pri výbere rozlíšenia treba zvoliť požadované.
- Po inštalácii je potrebný reštart počítača

4. Po reštarte počítača spustiť terminál a zadať príkazy (odstránenie nepotrebných súborov a adresárov):

- `sudo -s`
- `cd Desktop/`
- `rm open-vm-tools-2008.04.14-87182.tar.gz`
- `rm VMwareTools-6.0.2-59824.i386.rpm`
- `rm VMwareTools-6.0.2-59824.tar.gz`
- `rm -r open-vm-tools-2008.04.14-87182/`
- `rm -r vmware-tools-distrib/`

1.2 Inštalácia rcssservera 3D v Ubuntu 8.04 Hardy Heron

1. Aktivovať Universe a Multiverse (väčšinou je všetko nastavené správne, nastavenie zdrojových adries, odkiaľ sa sťahujú balíčky)

- `sudo gedit /etc/apt/sources.list`
- `sudo apt-get update`

2. Inštalácia doplnkov

- `sudo apt-get install g++ ruby1.9 ruby1.9-dev libode0-dev libboost-dev libsdl-dev libfreetype6-dev libdevil-dev autoconf automake1.9 libtool freeglut3-dev tetex-extra cvs xlibs-dev libtiff4-dev libslang1-dev libboost-thread-dev libmng-dev libpng12-dev`
- ak vám vypísalo chybu na "xlibs-dev" tak to zmeňte za "xlibs-static-dev"
- po odenterovaní sa bude ťahať cca 320MB z internetu!!!
- `sudo rm /usr/bin/ruby`
- `sudo ln -s /usr/bin/ruby1.9 /usr/bin/ruby`
- `sudo ln -s /usr/lib/libruby1.9.so /usr/lib/libruby.so`

3. Skontrolovať zdrojové kódy na source forge

- `cvs -d:pserver:anonymous@sserver.cvs.sourceforge.net:/cvsroot/sserver login`
- Ak bude pýtať heslo, dať enter (ak vypíše chybu o neexistujúcom cvspass, opakovať príkaz)
- `cvs -z3 -d:pserver:anonymous@sserver.cvs.sourceforge.net:/cvsroot/sserver co -P rcsoccersim/rcssserver3D`

4. Build a inštalácia servera

- `cd rcsoccersim/rcssserver3D/`
- `./bootstrap`
- `./configure`
- Ak vyhodí chybu kvôli ruby, tak sa musia odinštalovať balíky ruby, nainštalovať zo zdrojových kódov (4,3 MB) a opäť spustiť configure:
 1. `sudo apt-get remove libruby*`
 2. `sudo wget ftp://ftp.ruby-lang.org/pub/ruby/1.8/ruby-1.8.6-p114.tar.gz`
 3. ALEBO stiahnuť zdrojový kód ruby z <http://www.ruby-lang.org/en/downloads/>, momentálne <ftp://ftp.ruby-lang.org/pub/ruby/1.8/ruby-1.8.6-p114.tar.gz>
 4. `sudo tar -xzf ruby-1.8.6-p114.tar.gz`
 5. `cd ruby-1.8.6-p114/`
 6. `sudo ./configure --enable-shared`
 7. `sudo make`
 8. `sudo make install`
 9. `cd /usr/lib`
 10. `sudo ln -s /usr/local/lib/libruby.so.1.8`
 11. `sudo ldconfig`
 12. `cd /home/<nazov_konta>/rcsoccersim/rcssserver3D/`
 13. `./configure`
- `make`
- `sudo make install`
- `sudo gedit /etc/ld.so.conf`
- dopísať riadok `include /usr/local/lib` . Uložiť a zavrieť
- `sudo ldconfig`

5. Spustenie simulácie

- Spustiť terminál
- napísať príkaz: `simspark`

6. Pridanie testovacieho hráča

- Spustiť terminál
- napísať príkaz: `agentspark`

Postup inštalácie vývojového prostredia Eclipse Europa pre Ubuntu 8.04 Hardy Heron

1. Inštalácia Javy

- `sudo apt-get update`
- `sudo apt-get install sun-java6-jdk`
- pri inštalácii potvrdiť licenčné podmienky

2. Inštalácia prostredia (Eclipse Europa)

- Stiahnut Eclipse Europa pre C++
http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/20070927/eclipse-cpp-europa-fall-linux-gtk.tar.gz&url=http://ftp.sh.cvut.cz/MIRRORS/eclipse/technology/epp/downloads/release/20070927/eclipse-cpp-europa-fall-linux-gtk.tar.gz&mirror_id=302
- `cd /home/<nazov_konta>/Desktop/`
- `sudo tar -xzf eclipse-cpp-europa-fall-linux-gtk.tar.gz`
- spustiť eclipse (bez koncovky)

3. Upratanie balíkov

- `sudo rm /home/<nazov_konta>/Desktop/eclipse-cpp-europa-fall-linux-gtk.tar.gz`

1.3 Inštalácia rcservera 3D pod Mac OS X

V tejto časti opíšeme inštaláciu RoboCup 3D servera verzie 0.5.6 pod Mac OS X verzie 10.4.10. V systéme máme nainštalované vývojové prostredie Xcode.

1. Otvoríme si aplikáciu Terminal a prihlásime sa ako superuser:
 - `sudo -s`
2. Stiahneme SDL (Simple Directmedia Layer) knižnicu z adresy <http://libsdl.org/release/SDL-1.2.12.tar.gz>, rozbalíme, skompilujeme a nainštalujeme. Nainštaluje sa do adresára `/usr/local/`.
 - `tar -xzfv SDL-1.2.12.tar.gz`
 - `cd SDL-1.2.12`
 - `./configure`
 - `make`
 - `make install`

Ďalšie verzie knižnice sa nachádzajú na stránke

<http://www.libsdl.org>

3. Stiahneme knižnicu DevIL (Developer's Image Library) z adresy <http://prdownloads.sourceforge.net/openil/DevIL-1.6.8-rc2.tar.gz>, rozbalíme, skompilujeme a nainštalujeme:
 - `tar -xzfv DevIL-1.6.8-rc2.tar.gz`
 - `cd DevIL-1.6.8`
 - `./configure`
 - `make`
 - `make install`

Ďalšie verzie knižnice nájdeme na stránke

<http://openil.sourceforge.net>

4. Ďalej budeme potrebovať Boost knižnicu. Najjednoduchší spôsob ako ju nainštalovať je pomocou nástroja MacPorts, ktorý získame zo stránky <http://www.macports.org>. Po nainštalovaní MacPorts môžeme pokračovať v inštalácii knižnice Boost:
 - `port install boost`

5. Stiahneme knižnicu ODE (Open Dynamics Engine) z adresy <http://downloads.sourceforge.net/opende/ode-src-0.9.zip>, rozbalíme, skompilujeme a nainštalujeme:

- `unzip ode-src-0.9.zip`
- `cd ode-0.9`
- `./configure`
- `make`
- `make install`

Ďalšie verzie knižnice sú na stránke <http://www.ode.org>

6. Stiahneme RoboCup 3D server z adresy <http://downloads.sourceforge.net/sserver/rcsserver3d-0.5.6.tar.gz> a textúry s modelmi z adresy <http://downloads.sourceforge.net/sserver/rcsserver3d-0.5.6-data.tar.gz>.

Oba archívy rozbalíme a modely s textúrami premiestnime do adresára servera:

- `tar -xvzf rcsserver3d-0.5.6.tar.gz`
- `tar -xvzf rcsserver3d-0.5.6-data.tar.gz`
- `mv models rcsserver3d-0.5.6/`
- `mv textures rcsserver3d-0.5.6/`

Potom spustíme konfiguračný skript RoboCup 3D serveru:

- `cd rcsserver3d-0.5.6`
- `CPPFLAGS="-I/opt/local/include/" LDFLAGS="-L/opt/local/lib" ./configure`

Po skončení konfiguračného skriptu odstránime dva súbory:

- `rm app/coretest/Makefile app/scenetest/Makefile`
- `vi app/coretest/Makefile`

Vytvoríme súbor Makefile v adresári `./app/coretest/`

- `Vi app/coretest/Makefile`

Do novovytvoreného Makefile súboru zapíšeme¹:

- `all:`
`touch all`
- `install:`
`touch install`

Novovytvorený Makefile uložíme a skopírujeme:

- `cp app/coretest/Makefile app/scenetest/Makefile`

Teraz môžeme skompilovať RoboCup 3D server:

- `make`
- `make install`

7. Funkčnosť servera overíme jeho spustením

- `export DYLD_LIBRARY_PATH="/usr/local/lib/rcsserver3d"`
`./app/simspark/simspark`

Môžeme pridať aj vzorového hráča `agentspark` na ihrisko

¹ Riadky začínajúce slovami `touch` sú odsadené tabulátorom

- ```
export
LD_LIBRARY_PATH="/usr/local/lib:/usr/local/lib/rcssserver3d"
./app/agentspark/agentspark
```



## 2. Hráč Sirius

V tejto časti opíšeme hráča Sirius.

### 2.1 Stiahnutie hráča a adresárová štruktúra projektu

Celý projekt hráča je možné stiahnuť z SVN nášho tímu :

- `svn://labss2.dcs.elf.stuba.sk:3691/hlavicka`

Projekt sa takisto nachádza na priloženom CD médiu.

Hlavný adresár hráča je /trunk a obsahuje nasledovné podadresáre

- trunk/build/ – do tohto adresára sa ukladajú skompilované zdrojové súbory, ktoré sú uložené v adresári prislúchajúcemu operačnému systému (linux, macosx, windows). Nachádza sa tu aj výsledný binárny súbor agenta.
- trunk/download/ – obsahuje projekty tretích strán, ktoré sme použili vo svojom projekte.
- trunk/include/ – obsahuje hlavičky súborov tretích strán použitých v projekte. Hlavičky spoločné pre všetky operačné systémy sa nachádzajú priamo v adresári include, špecifické hlavičky v samostatných adresároch prislúchajúcemu operačnému systému (linux, macosx, windows)
- trunk/lib/ - má podobnú funkciu ako adresár trunk/include. Obsahuje lib knižnice.
- trunk/projects/ - obsahuje projektové súbory vo viacerých vývojových prostrediach pre rôzne platformy. Použité prostredia sú: Visual Studio 6.0 (Windows), Visual Studio 2005 (Windows), Visual Studio 2008 (Windows), Eclipse (Linux), Xcode (Mac OS).
- trunk/src/ - obsahuje zdrojové kódy hráča
- trunk/temp/ - pomocný (resp. intermediate) adresár, ktorý sa používa pri kompilácii
- trunk/test/ - adresár slúžiaci pre testovacie účely
- trunk/utilities/ - adresár obsahujúci programy, ktoré sa používajú pri tvorbe hráča

Ďalej podrobnejšie opíšeme štruktúru adresára trunk/src/ , v ktorom sa nachádzajú zdrojové kódy hráča:

- /src/behaviour – tu sú zdrojové kódy, ktoré implementujú zručnosti a správanie hráča. Podľa navrhutej architektúry sú rozdelené na Low skills, Middle skills a High skills. Pre každú zručnosť je vytvorená trieda, ktorá ju implementuje.
- /src/communication – obsahuje implementáciu komunikácie hráča so serverom
- /src/config – načítanie nastavenia hráča z konfiguračného súboru. Súbor soccertypes obsahuje názvy kĺbov hráča a hracích módov
- /src/logger – zdrojové súbory loggeru, ktorý sa používa hlavne počas implementácie na logovanie zadaných parametrov
- /src/parser – implementácia parsera správ zo serveru
- /src/playermodel – obsahuje model hráča, efekty a perceptory pre pohyb kĺbov a tiež implementáciu ovládania kĺbov. Ďalej obsahuje implementáciu gyroskopu a perceptoru FRP (force resistance perceptor)

- /src/thread – implementácia vlákien
- /src/tinyxml – zdrojové súbory knižnice pre prácu s xml súbormi
- /src/worldmodel – obsahuje implementáciu modelu sveta hráča
- /src/agent.xml – konfiguračný súbor hráča
- /src/Agent.cpp, Agent.h – tieto súbory obsahujú implementáciu hlavnej triedy agenta a hlavnú slučku programu.
- /src/main.cpp – zdrojový súbor s funkciou main

## 2.2 Kompilácia a spustenie hráča

Projektové súbory hráča sa nachádzajú v adresári trunk/projects/. Projekty v uvedených vývojových prostrediach sú vždy aktualizované k poslednej verzii hráča. Pre skompilovanie hráča treba otvoriť projekt vo vybranom vývojovom prostredí a zvoliť možnosť „build“. Skompilované súbory a taktiež výsledný binárny súbor sa uložia do adresára trunk/build/.

### Konfiguračný súbor hráča

Pred spustením hráča treba nastaviť niektoré dôležité parametre v konfiguračnom súbore hráča trunk/src/agent.xml. Tento konfiguračný súbor treba prekopírovať (ak sa tam ešte nenachádza) do adresára, z ktorého budeme spúšťať hráča. Pri použití vývojového prostredia VS 2005 a 2008 sa po kompilácii spustí post-script (trunk/projects/VS2008/Sirius/copy\_xml.bat), ktorý prekopíruje upravený xml súbor z trunk/src/agent.xml do adresára, kde sa nachádza projekt.

Ďalej uvedieme obsah konfiguračného súboru (iba časti potrebné pre spustenie hráča):

- **Connection:** Obsahuje informácie o komunikácii. Jeho nasledovné atribúty sú:
  - IP: IP adresa servera (v nainštalovanom Ubuntu vo virtuálnej mašine zistíme IP adresu rozhrania v konzole cez príkaz ifconfig)
  - Port: port servera, default je 3100
  - Type: typ komunikácie (tcp alebo udp), odporúčame zvoliť tcp
  - UseThreads: použitie vlákien (yes alebo no)
- **Player:** Obsahuje informácie o hráčovi
  - Team: názov tímu
  - Number: číslo hráča
- **World:** obsahuje informácie o hracej ploche (svete)
  - Ball: obsahuje atribút radius, ktorý nastavuje veľkosť lopty, odporúčame nechať prednastavenú hodnotu 0.5, ktorá je kompatibilná s poslednou verziou servera

### Spustenie hráča

Hráč sa spúšťa zo samotného vývojového prostredia alebo pomocou binárneho súboru, ktorý sa vygeneruje (napr. pre Windows) do trunk/build/windows/Debug (ak sa jedná o Debug verziu) alebo do trunk/build/windows/Release (ak ide o Release verziu). V adresároch Debug aj Release musí byť umiestnený súbor pthreadVC2.dll, ktorý obsahuje metódy pre použitie vlákien.

Pred spustením hráča treba samozrejme spustiť simulačný server. Keď nám server beží, hráča spustíme zadaním „agent.exe“ so zvoleným prepínačom. Popis prepínačov je uvedený nižšie.

Po pripojení sa hráč začne vykonávať implementovanú zručnosť, až keď je simulácia v móde KickOff (v serveri treba stlačiť „K“). Implementované zručnosti zahŕňajú jednoduché pohyby,

ako je podrep, vstávanie hráča zo zeme a jednoduchá chôdza. Všetky implementované zručnosti sú v beta verzii vo fáze testovania. Boli vytvorené predovšetkým na otestovanie a demonštrovanie implementovanej architektúry hráča a základných funkcií pre ovládanie kĺbov.

### Parametre hráča

Pre beta verziu hráča boli vytvorené rôzne prepínače, ktorými sa aktivuje konkrétna implementovaná vlastnosť hráča.

- -h : help, výpis a opis všetkých argumentov
- -c : nastavenie cesty ku konfiguračnému XML súboru
- --xmltest : XML test pre TinyXML
- --skilltest walk – spustí sa implementovaná chôdza hráča
- --skilltest crouch – hráč robí v cykle drepy
- --skilltest standup – implementácia vstávania

Príklad spustenia hráča so zvolenou zručnosťou chôdze:

- agent.exe --skilltest walk