

Prog2, Java tárgyalás

Magasszintű programozási nyelvek 2 mérnök informatikus BSc előadás

Dr. Bátfai Norbert

egyetemi adjunktus

<http://www.inf.unideb.hu/~nbatfai/>

Debreceni Egyetem, Informatikai Kar,
Információ Technológia Tanszék

batfai.norbert@inf.unideb.hu

Skype: batfai.norbert

Prog2_2.ppt, v.: 0.0.5, 2011. 11. 27.

<http://www.inf.unideb.hu/~nbatfai/>

<http://nehogy.fw.hu/>

Az óra blogja: <http://progater.blog.hu/>

A Nokia Ovi store-ban is elérhető: <http://store.ovi.com/content/100794>

Felhasználási engedély

Bátfai Norbert

Debreceni Egyetem, Informatikai Kar, Információ Technológia Tanszék
<nbatfai@inf.unideb.hu, nbatfai gmail com>

Copyright © 2011 Bátfai Norbert

E közlemény felhatalmazást ad önnek jelen dokumentum sokszorosítására, terjesztésére és/vagy módosítására a Szabad Szoftver Alapítvány által kiadott GNU Szabad Dokumentációs Licenc 1.2-es, vagy bármely azt követő verziójának feltételei alapján. Nem változtatható szakaszok: A szerzőről.

Címlap szövegek: Programozó Páternoszter, Bátfai Norbert, Gép melletti fogyasztásra.

Hátlap szövegek: GNU Jávácska, belépés a gépek mesés birodalmába.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with the Invariant Sections being: A szerzőről, with the Front-Cover Texts being: Programozó Páternoszter, Bátfai Norbert, Gép melletti fogyasztásra, and with the Back-Cover Texts being: GNU Jávácska, belépés a gépek mesés birodalmába.

Kapcsoldó videók, videómagyarázatok és blogok

Az előadás és a labor posztja ez(ek)

1) Nem mindig a jobbik csapat nyer...*:

[http://progpater.blog.hu/2011/09/16/nem mindig a jobbik csapat nyer](http://progpater.blog.hu/2011/09/16/nem_mindig_a_jobbik_csapat_nyer)

We're a unit, not a one-man show*:

[http://progpater.blog.hu/2011/09/23/we re a unit not a one man show](http://progpater.blog.hu/2011/09/23/we_re_a_unit_not_a_one_man_show)

2) FerSML - Football Computing:

[http://fersml.blog.hu/2011/06/12/football computing](http://fersml.blog.hu/2011/06/12/football_computing)

Célok és tartalom

Előadás

- a) Néhány osztály kapcsolatának megtervezése, Software Engineering fogalma, OO tervezés, modellezés, RUP, UML
- b) XML dokumentum jól formázottsága és érvényessége, RELAX NG
- c) A RoboCup 2D szimulációs liga
- d) Java fejlesztés: Apache Maven
- e) A Java interfész és csomag fogalma
- f) Java nyelvi bevezetés: öröklődés, interfészek, absztrakt osztályok, metódusok.

Labor (2 alkalomra tervezve)

- a) a Javát tanítók példáinak felélesztése
EXOR kódolt szöveg megfejtése Javában
- b) Atan alapú robotfoci:
Aranylabor FC, Csak a foci FC, Büntető FC, Kapuba FC, Foci iszony FC, Ping-pong FC, Mágnes FC, Kékhalál FC

Laborkártyák

- a) Interfész kártyák, kódolási konvenciók kártyák

Otthoni opcionális feladat

- a) Saját Atan alapú RCSS csapat fejlesztése

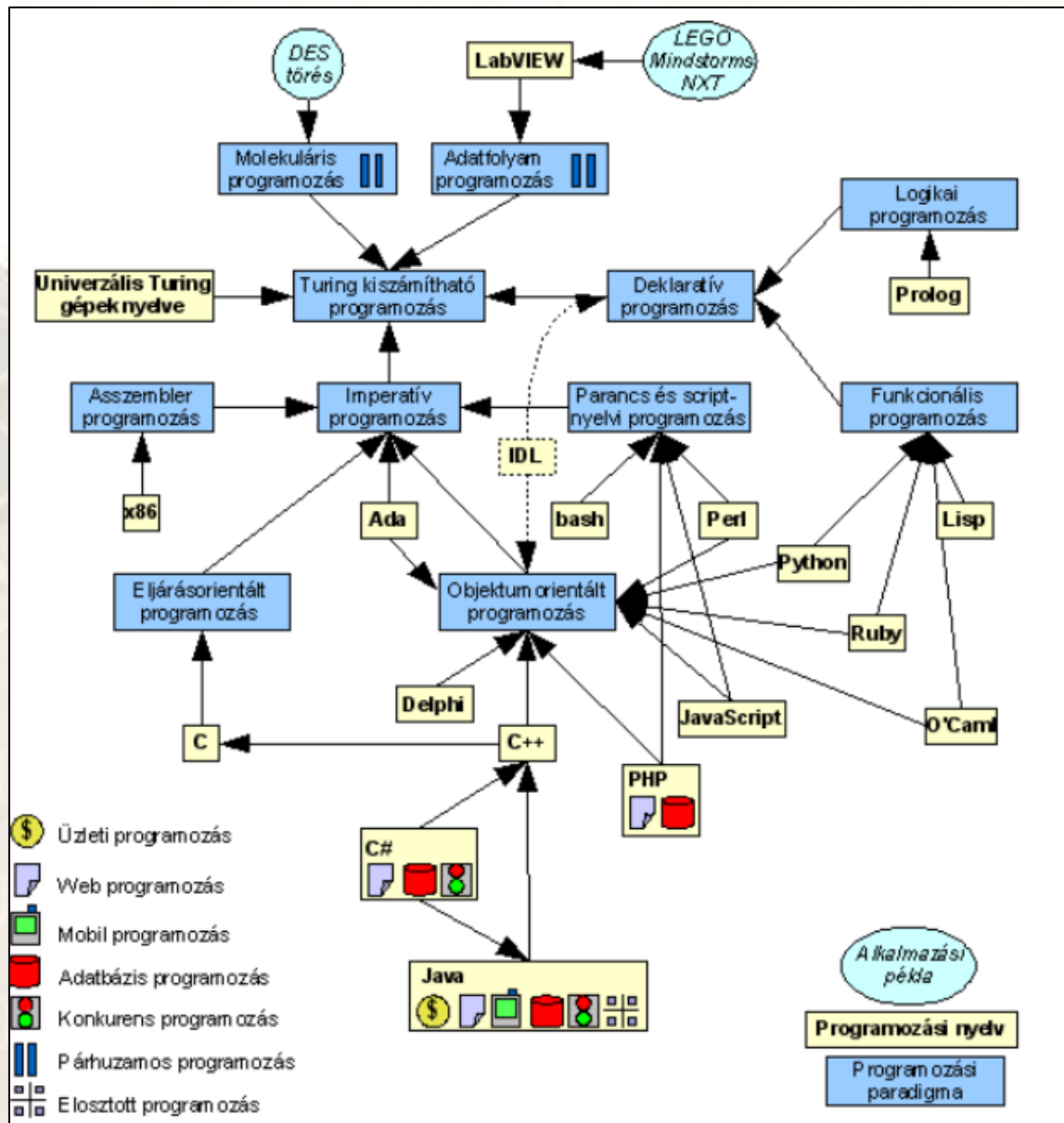
Minimális gyakorlati cél

- a) A hallgató el tudjon készíteni (verbális terv, programozás, fordítás, futtatás) egy Atan alapú robotfoci csapatot.
- b) Le tudja fordítani és tudja futtatni a FerSML platform PublicResourceFC programját, illetve a kurzus Aranycsapat FC csapatát (Maven projektek).
- c) Tudjon egyszerű esetekben UML osztálydiagramot tervezni-rajzolni, vagy ilyen generálni meglévő Java forrásokból, például az ArgoUML-el: <http://argouml.tigris.org/>

Minimális elméleti cél

- 1) A hallgató néhány szóban tudja jellemezni a főbb szoftverfejlesztési modelleket.
- 2) A hallgató néhány osztályból és interfészből álló rendszert meg tudjon tervezni (UML osztálydiagram alakjában le tudja rajzolni).
- 3) A hallgató tudjon beszélni XML dokumentum jól formázottságáról és érvényességéről, tudjon egyszerű nyelvtant leírni RELAX NG-vel.
- 4) Ismerje a Maven használatát az előadásban használt két példa (FerSML, illetve RCSS) kapcsán.
- 5) Java nyelv kapcsán: interfészek, csomagok.

Java (eredet) sokadik ism.



- Imperatív
- Objektum orientált és eljárás orientált paradigma mentén

Aki Javában programoz OO programoz, de az egyszerű típusok, változó fogalom, kifejezések mint C-ben.

Didaktika

„az egyszerű típusok, változó fogalom, kifejezések mint C-ben” – ezért sem fogunk olyan fóliát találni, amin ezekkel foglalkozunk, hanem lásd az első előadás kötelező olvasmányát!

Az Java program OO szerkezete

Milyen elemekből építkezünk a Java program készítésekor?

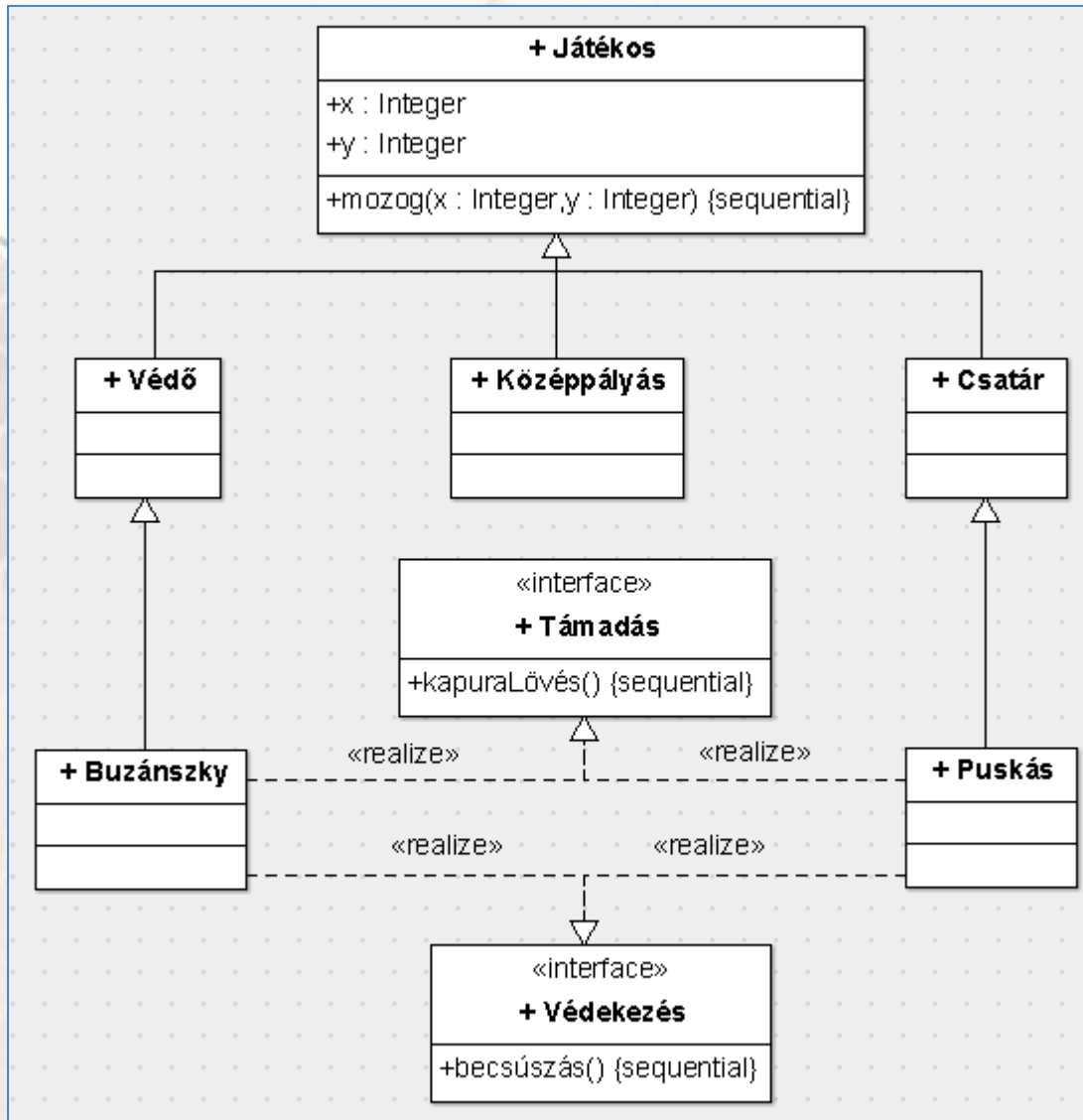
- a) Osztályok
- b) Öröklődés
- c) Interfészek
- d) Csomagok

Interfészek

Mik az interfészek?

Mire jók?

Interfészek – egy példa



Puskás puskás =
new Puskás();

...

Használat az
implementáción
keresztül

puskás.kapuraLövés();

Interfészek – új absztrakciós szint

TÁMADÁS / VÉDEKEZÉS

Játékosok

Védők

Támadók

Puskás

Középpályások

Interfészek – néhány konkrét példa

FerSML

<http://sourceforge.net/projects/footballerml/>

PublicResourceFC\src\main\java\hu\javacska\sfa\src\main\java\hu\javacska\sfa\

```
package hu.javacska.sfa;

/**
 * The Automated Soccer Applet.
 *
 * @author      Norbert Bátfai
 * @version     0.0.5
 */
public class SFApplet extends java.applet.Applet
    implements java.awt.event.KeyListener, FootballField,
        ViewableField, SimulationType {

    /** A Football Match that must display in this Applet. */
    FootballMatch footballMatch;
```

Interfészek – java.awt.event.KeyListener

Java SE 7 API doksi

<http://download.oracle.com/javase/7/docs/api/java/awt/event/KeyListener.html>

Since:

1.1

See Also:

[KeyListenerAdapter](#), [KeyEvent](#), [Tutorial: Writing a Key Listener](#)

Method Summary

Methods

Modifier and Type	Method and Description
void	keyPressed (KeyEvent e) Invoked when a key has been pressed.
void	keyReleased (KeyEvent e) Invoked when a key has been released.
void	keyTyped (KeyEvent e) Invoked when a key has been typed.

Interfészek – java.awt.event.KeyListener

c:\Program Files\Java\jdk1.7.0\src.zip

java\awt\event\KeyListener.java

```
public interface KeyListener extends EventListener {

    /**
     * Invoked when a key has been typed.
     * See the class description for {@link KeyEvent} for a definition of
     * a key typed event.
     */
    public void keyTyped(KeyEvent e);

    /**
     * Invoked when a key has been pressed.
     * See the class description for {@link KeyEvent} for a definition of
     * a key pressed event.
     */
    public void keyPressed(KeyEvent e);

    /**
     * Invoked when a key has been released.
     * See the class description for {@link KeyEvent} for a definition of
     * a key released event.
     */
    public void keyReleased(KeyEvent e);
}
```

Interfészek – java.awt.event.KeyListener

c:\Program Files\Java\jdk1.7.0\src.zip

java\awt\event\KeyListener.java

Since:

1.1

See Also:

[KeyAdapter](#), [KeyEvent](#), [Tutorial: Writing a Key Listener](#)

Method Summary

Methods

Modifier and Type	Method and Description
void	keyPressed (KeyEvent e) Invoked when a key has been pressed.
void	keyReleased (KeyEvent e) Invoked when a key has been released.

```
* @author Carl Quinn
*
* @see KeyAdapter
* @see KeyEvent
* @see <a href="http://java.sun.com/docs/books/tutorial/post1.0/ui/keylistener.html">Tutorial:
*
* @since 1.1
*/
```


Interfészek – java.awt.event.KeyAdapter

c:\Program Files\Java\jdk1.7.0\src.zip

java\awt\event\KeyAdapter.java

```
public abstract class KeyAdapter implements KeyListener {
    /**
     * Invoked when a key has been typed.
     * This event occurs when a key press is followed by a key release.
     */
    public void keyTyped(KeyEvent e) {}

    /**
     * Invoked when a key has been pressed.
     */
    public void keyPressed(KeyEvent e) {}

    /**
     * Invoked when a key has been released.
     */
    public void keyReleased(KeyEvent e) {}
}
```

Interfészek – néhány konkrét példa

FerSML

<http://sourceforge.net/projects/footballerml/>

PublicResourceFC\src\main\java\hu\javacska\sfa\src\main\java\hu\javacska\sfa\

```
public void keyPressed(java.awt.event.KeyEvent e) {  
  
    if (e.getKeyCode() == java.awt.event.KeyEvent.VK_T) {  
        tacticsView = !tacticsView;  
    } else if (e.getKeyCode() == java.awt.event.KeyEvent.VK_H) {  
        homePlayerForceView = !homePlayerForceView;  
    } else if (e.getKeyCode() == java.awt.event.KeyEvent.VK_J) {  
        sumPlayerForceView = !sumPlayerForceView;  
    } else if (e.getKeyCode() == java.awt.event.KeyEvent.VK_K) {  
        } else if (e.getKeyCode() == java.awt.event.KeyEvent.VK_C) {  
            sumSocceralBoxForceView = !sumSocceralBoxForceView;  
        } else if (e.getKeyCode() == java.awt.event.KeyEvent.VK_G) {  
            footballMatch.setGivenSituation();  
        }  
    }  
  
} public void keyTyped(java.awt.event.KeyEvent billentyuEsemeny) {  
}  
  
} public void keyReleased(java.awt.event.KeyEvent billentyuEsemeny) {  
}
```

Interfészek – néhány konkrét példa

FerSML

<http://s>

Public

public

```
public interface SimulationType {  
    /*  
    public static final int FOOTBALL_MATCH = 0;  
    public static final int MANY_FOOTBALL_MATCH_SIM = 10;  
    public static final int FOOTBALL_WORLD_CUP = 100;  
    public static final int MANY_FOOTBALL_WORLD_CUP_SIM = 1000;  
    */  
    *  
    *  
    public static final String FootballMatch = "FOOTBALL_MATCH";  
    *  
    *  
    public static final String ManyFootballMatchSim = "MANY_FOOTBALL_M  
    /*  
    public static final String AllFormationsSim = "ALL_FORMATIONS_SIM"  
    fi public static final String FootballWorldCup = "FOOTBALL_WORLD_CUP"  
    /*  
    public static final String FootballWorldCupSim = "FOOTBALL_WORLD_C  
    fi public static final String ManyFootballWorldCupSim = "MANY_FOOTBAL  
    public static final String ManySimFromSemiFinals = "MANY_SIM_FROM_  
    public static final String[] types = {  
        public interface ViewableField {  
            FootballMatch,  
            ManyFootballMat  
            AllFormationsSim  
            FootballWorldCup  
            FootballWorldCup  
            ManyFootballWor  
            ManySimFromSemi  
            void draw();  
            void commentator(String msg);  
            void setSquadColours(int home, int away);  
            void setTeamLogo(int teamIdx, java.awt.Image logo);  
        }  
    };  
}
```

Interfészek – néhány konkrét példa

FerSML

<http://sourceforge.net>

PublicResourceFC

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Uruguay-Ghana

- S: screenshot
- R: home socceral
- B: away socceral
- E: sum socceral
- T: tactics lines
- P: players
- L: the ball
- Y: home socceral box
- X: away socceral box
- C: sum socceral box
- H: home players
- K: away players
- J: sum players

GNU GPL v3

GPLv3 FerSML GPLv3 GPLv3 FerSML GPLv3 GPLv3 FerSML GPLv3 GPLv3 FerSML

The 367-th match
4-3-3 - 4-3-3
189 / 126
51

2:2 0:0 Uruguay vs. Ghana

Pass Uruguay 1 -> Uruguay 3

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Uruguay-Ghana

2:1 0:0 Uruguay vs. Ghana

Góll Uruguay 3

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

GPLv3 FerSML GPLv3 GPLv3 FerSML GPLv3 GPLv3 FerSML

The 140-th match
4-3-3 - 4-3-3
76 / 44
19

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Interfészek – néhány konkrét példa

FerSML

<http://sourceforge.net/projects/footballerml/>

PublicResourceFC\src\main\java\hu\javacska\sfa\src\main\java\hu\javacska\sfa\

```
public interface ViewableField {  
  
    void draw() ;  
    void commentator(String msg) ;  
    void setSquadColours(int home, int away) ;  
    void setTeamLogo(int teamIdx, java.awt.Image logo) ;  
  
}
```

src\main\java\hu\javacska\sfa\src\main\java\hu\javacska\sfa\SFApplet.java

```
/** draw() will be callbacked by the object footballMatch. */  
public void draw() {  
  
    repaint() ;  
  
}
```

```
/** commentator() will be callbacked by the object footballMatch. */  
public void commentator(String s) {  
  
    commentatorMsg = s ;  
    repaint() ;  
  
}
```

Interfészek – néhány konkrét példa

FerSML

<http://sourceforge.net/projects/fer-sml/>

Public Resource

pac

```
/**  
 *  
 *  
 *  
 *  
 */
```

```
if (hu
```

```
foot
```

```
} else
```

```
foot
```

```
} else
```

```
foot
```

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Brazil-Argentina

GNU GPL v3

- S: screenshot
- R: home soccerall
- B: away soccerall
- E: sum soccerall
- T: tactics lines
- P: players
- L: the ball
- Y: home soccerall box
- X: away soccerall box
- C: sum soccerall box
- H: home players
- K: away players
- I: sum players

0:0 0:21 Brazil vs. Argentina

Argentina (10) beleért

Public Resource Football Computing for FerSML 0.0.14 - 2010.09.12

Labor - egyetemi feladatok

- a) Saját robotfoci csapat, ami eredményesen játszik egy adott csapat ellen.
- b) FerSML platformbeli fejlesztés: egyénileg egyeztetve.
- c) Kiterjesztett Jávácska Kupa.

Labor - céges feladatok

<http://www.inf.unideb.hu/~nbatfai/MernokInformatikusLabor.docx>

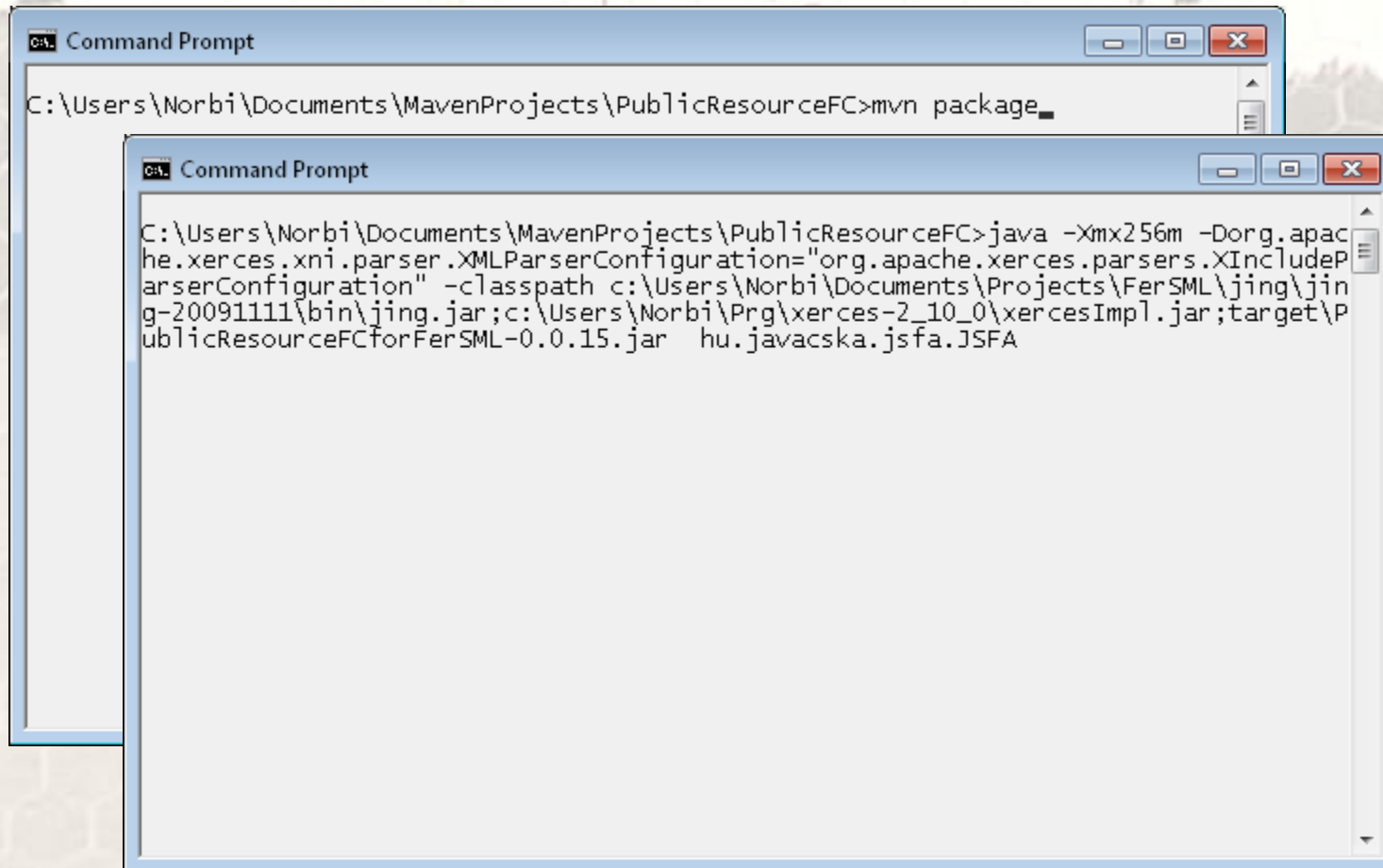
<http://www.inf.unideb.hu/~nbatfai/MernokInformatikusLabor.pdf>

Labor - FerSML

FerSML

<http://sourceforge.net/projects/footballerml/>

PublicResourceFC\src\main\java\hu\javacska\sfa\src\main\java\hu\javacska\sfa\



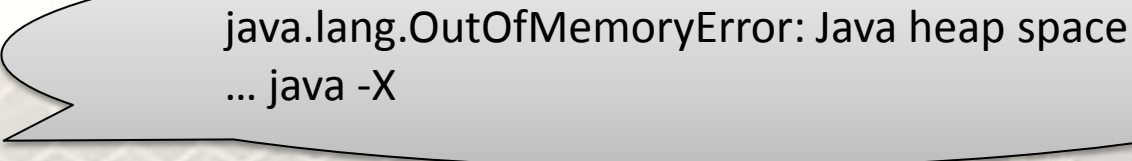
```
Command Prompt
C:\Users\Norbi\Documents\MavenProjects\PublicResourceFC>mvn package_

Command Prompt
C:\Users\Norbi\Documents\MavenProjects\PublicResourceFC>java -Xmx256m -Dorg.apache.xerces.xni.parser.XMLParserConfiguration="org.apache.xerces.parsers.XIncludeParserConfiguration" -classpath c:\Users\Norbi\Documents\Projects\FerSML\jing\jing-20091111\bin\jing.jar;c:\Users\Norbi\Prg\xerces-2_10_0\xercesImpl.jar;target\PublicResourceFCforFerSML-0.0.15.jar hu.javacska.jsfa.JSFA
```

Labor - FerSML

```
C:\Users\Norbi\Documents\MavenProjects\PublicResourceFC>
```

```
java -Xmx256m
```



java.lang.OutOfMemoryError: Java heap space
... java -X

```
-Dorg.apache.xerces.xni.parser.XMLParserConfiguration=  
"org.apache.xerces.parsers.XIncludeParserConfiguration"
```

```
-classpath
```

```
c:\Users\Norbi\Documents\Projects\FerSML\jing\jing-20091111\bin\jing.jar;  
c:\Users\Norbi\Prg\xerces-2_10_0\xercesImpl.jar;  
target\PublicResourceFCforFerSML-0.0.15.jar
```

```
hu.javacska.jsfa.JSFA
```

- Z
- Z
- Z
- Z
- Z
- Z
- Z
- Z

First, you must to validate avatars! The jing.jar (jing-20091111) required for that purpose is downloadable from <http://code.google.com/p/jing-trang/> or choose the button labeled "Start without validation" if You don't want to use FerSML avatar files. But in this case only the default values will be used.

Further information can be found in <http://arxiv.org/abs/1004.2003>

Copyright (C) 2010, Norbert Batfai, <nbatfai@gmail.com>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

- Z
-

First, you must to validate avatars! The jing.jar (jing-20091111) required for that purpose is downloadable from <http://code.google.com/p/jing-trang/> or choose the button labeled "Start without validation" if You don't want to use FerSML avatar files. But in this case only the default values will be used.

Further information can be found in <http://arxiv.org/abs/1004.2003>

Copyright (C) 2010, Norbert Batfai, <nbatfai@gmail.com>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

YES, I agree the terms of this licence.

- FOOTBALL_MATCH
- MANY_FOOTBALL_MATCH_SIM
- ALL_FORMATIONS_SIM
- FOOTBALL_WORLD_CUP
- FOOTBALL_WORLD_CUP_SIM
- MANY_FOOTBALL_WORLD_CUP_SIM
- MANY_SIM_FROM_SEMI_FINALS

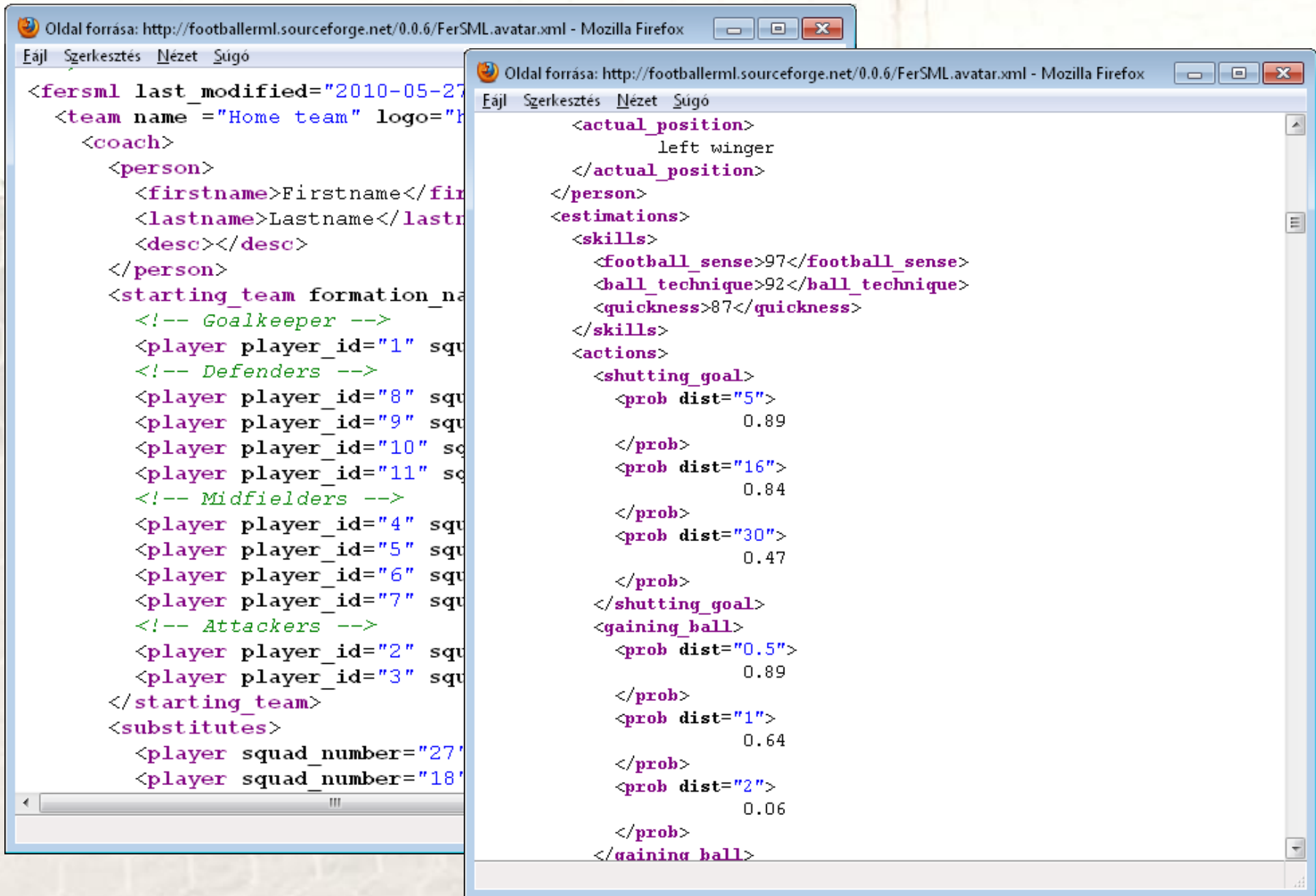
Home team

Away team

of World Cups



Szurkolói avatárok



```
<fersml last_modified="2010-05-27"
  <team name="Home team" logo="..."
    <coach>
      <person>
        <firstname>Firstname</firstname>
        <lastname>Lastname</lastname>
        <desc></desc>
      </person>
    <starting_team formation_name="..."
      <!-- Goalkeeper -->
      <player player_id="1" squad_number="..."
      <!-- Defenders -->
      <player player_id="8" squad_number="..."
      <player player_id="9" squad_number="..."
      <player player_id="10" squad_number="..."
      <player player_id="11" squad_number="..."
      <!-- Midfielders -->
      <player player_id="4" squad_number="..."
      <player player_id="5" squad_number="..."
      <player player_id="6" squad_number="..."
      <player player_id="7" squad_number="..."
      <!-- Attackers -->
      <player player_id="2" squad_number="..."
      <player player_id="3" squad_number="..."
    </starting_team>
    <substitutes>
      <player squad_number="27"
      <player squad_number="18"

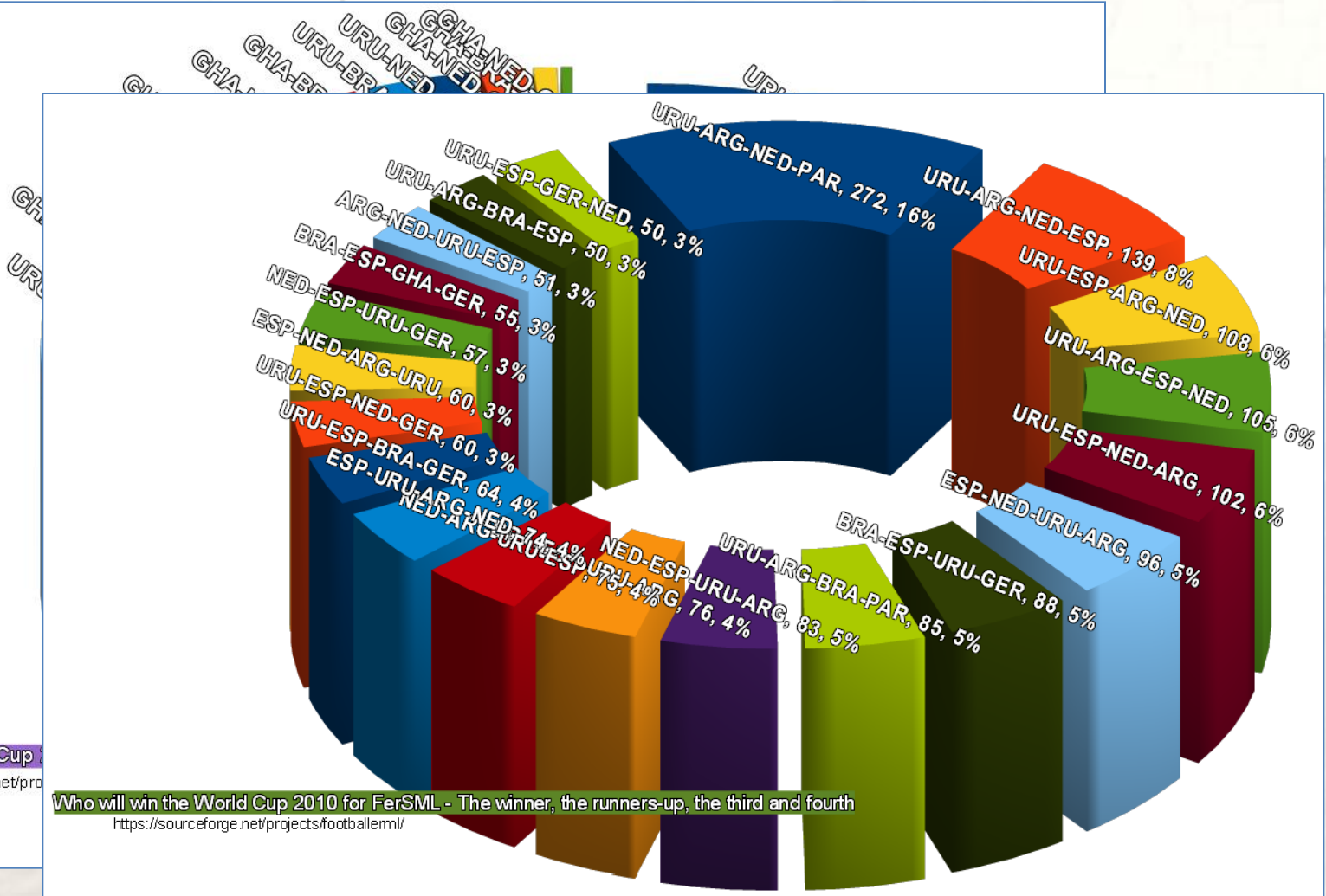
</fersml>
```

```
    <actual_position>
      left winger
    </actual_position>
  </person>
  <estimations>
    <skills>
      <football_sense>97</football_sense>
      <ball_technique>92</ball_technique>
      <quickness>87</quickness>
    </skills>
    <actions>
      <shutting_goal>
        <prob dist="5">
          0.89
        </prob>
        <prob dist="16">
          0.84
        </prob>
        <prob dist="30">
          0.47
        </prob>
      </shutting_goal>
      <gaining_ball>
        <prob dist="0.5">
          0.89
        </prob>
        <prob dist="1">
          0.64
        </prob>
        <prob dist="2">
          0.06
        </prob>
      </gaining_ball>
    </actions>
  </estimations>
</fersml>
```

FerSML

Footballer and Football Simulation Markup Language

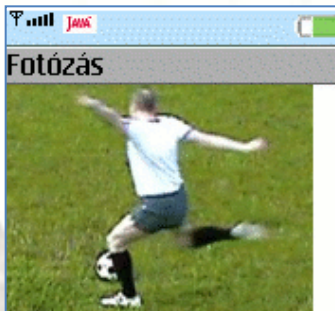
Probléma: a valódi futball szimulálható?



Who will win the World Cup
<https://sourceforge.net/pro>

Who will win the World Cup 2010 for FerSML - The winner, the runners-up, the third and fourth
<https://sourceforge.net/projects/footballerml/>

FerSML



Vissza Kattint

Piramis számítá
piramis) adják.

Automated Soccer Applet 0.0.11 for FerSML (from the "Soccer Game 4u OSE", http://javacska.sourceforge.net/mobile_game)

Magyar Svéd
Brazíl Angol
Francia Német
Amerikai Olasz
Magyar Brazíl
Francia Amerikai

SFA11 0.0.1 for FerSML (from the "Soccer Game 4u OSE", http://javacska.sourceforge.net/mobile_game)

Magyar-Francia
The 2575-th matches

GNU GPL v3

S: screenshot
R: home soccerball
B: away soccerball
E: sum soccerball
T: tactics lines
P: players
L: the ball
Y: home soccerball box
X: away soccerball box
C: sum soccerball box

JÁTÉKOK EUROSMOBIL JÁTÉKOK EUROSMOBIL JÁTÉKOK EUROSMOBIL JÁTÉKOK EUROSMOBIL JÁTÉKOK EUROSMOBIL JÁTÉKOK EUROSMOBIL JÁTÉKOK

Automated Soccer Applet 0.0.11 for FerSML (from the "Soccer Game 4u OSE", http://javacska.sourceforge.net/mobile_game)

1:0 0:0 Magyar vs. Francia

Átadás Kék 6 -> Kék 5

SFA11 0.0.1 for FerSML (from the "Soccer Game 4u OSE", http://javacska.sourceforge.net/mobile_game)

A „piramis” jeletése



5-3-2: a piramis (Cambridge pyramid) az 1870-es 1880-as évekből...

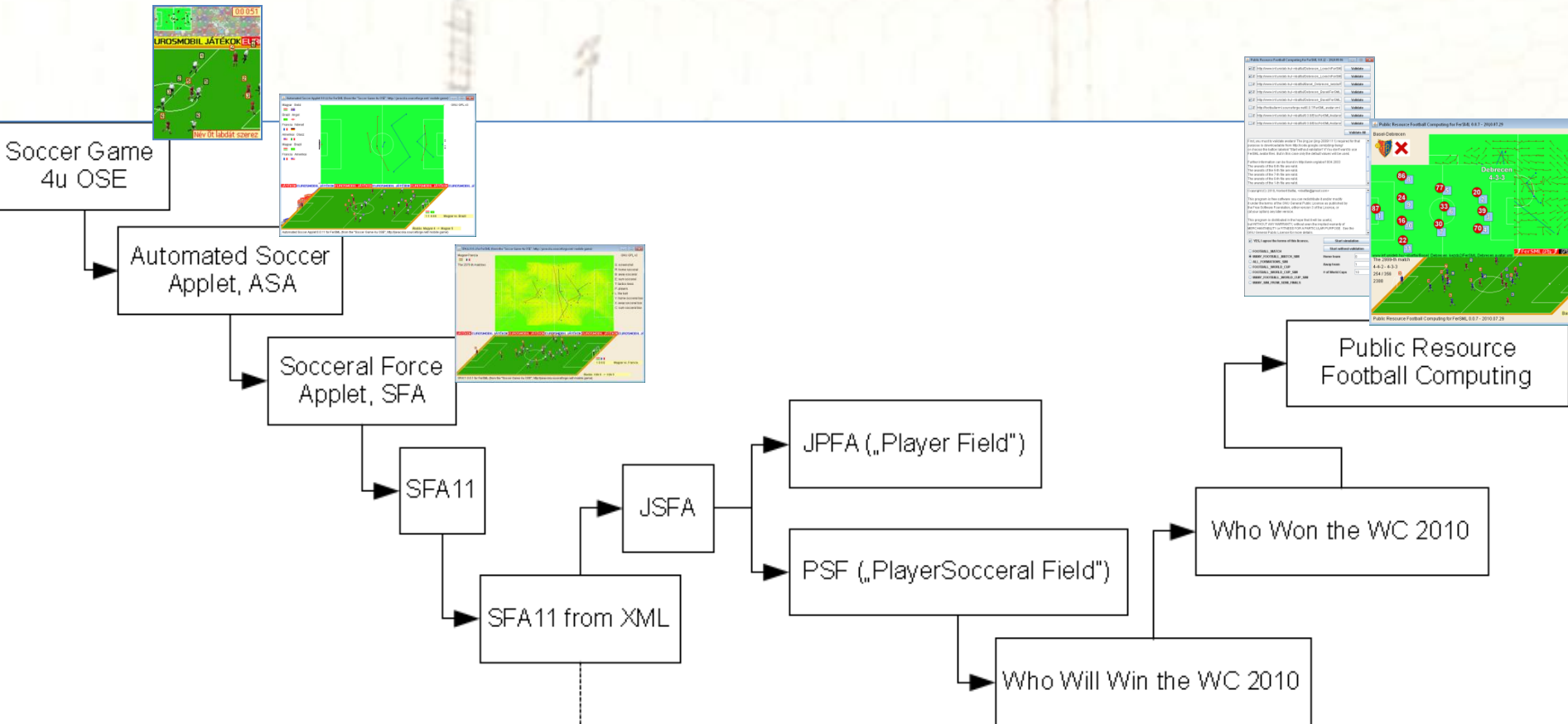
- Csanádi Árpád: Hungarian coaching manual, Soccer, Corvina, Budapest 1965.
- Jonathan Wilson: Inverting the Pyramid: a History of Football Tactics, Orion Books, 2008

További piramisok



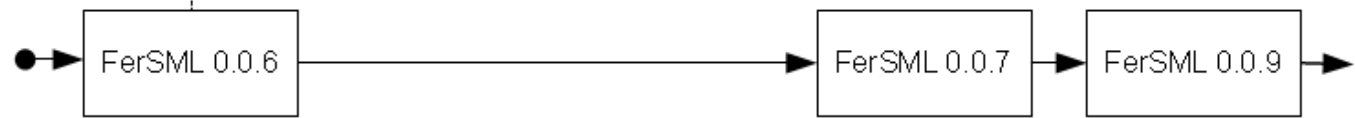
WM (esetünkben 3-2-2-3), a „6-3” az Aranycsapat felállása
4-2-4

FerSML



Softwares

FerSML (the XML language)



Mire (lesz, lenne) jó?

a futball szimuláció egy használati esete

Tekintsük meg a *N. Bátfai, E. Bátfai (2010), Public Resource Computing in European Football (submitted)* beküldött cikk példáját, mint egy használati esetet. Itt a BL 3. selejtezőkörének Debreceni VSC - FC Basel (0-2) és FC Basel – Debreceni VSC (3-1) mérkőzéséről van szó.

Szimuláció

egy használati eset

Some statistical data collected by UEFA and us, where "SN" means squad number, TGB - tackles with gaining the ball, ATW - attempts on target and wide, LSR - long solo runs.

Debreceni VSC vs. FC Basel						
SN	UEFA data			Our own results		
	TGB	ATW	LSR	TGB	LSR	ATW
D-21	-	0	-	1	0	1
D-24	-	1	-	0	0	0
D-16	-	0	-	0	0	0
D-22	-	2	-	0	0	1
D-20	-	0	-	2	0	2
D-33	-	2	-	1	1	2
D-30	-	1	-	0	0	1
D-77	-	1	-	3	0	2
D-39	-	1	-	2	0	2
D-10	-	0	-	0	0	0
D-15	-	0	-	1	0	1
B-20	-	0	-	3	0	0
B-4	-	0	-	0	0	0
B-19	-	1	-	0	0	1
B-22	-	1	-	2	0	1
B-34	-	0	-	0	0	1
B-24	-	0	-	0	0	0
B-8	-	0	-	0	0	1
B-17	-	2	-	0	0	2
B-31	-	1	-	0	0	0
B-13	-	2	-	0	0	2
B-14	-	2	-	0	0	3
B-30	-	0	-	1	0	0

FerSML szimuláció

egy használati eset



FerSML szimuláció

egy használati eset

Public Resource Football Computing for FerSML 0.0.7														
4x3x3			4x4x2			4x3x2x1			3x4x3			4x2x3x1		
BW	DW	DG	BW	DW	DG	BW	DW	DG	BW	DW	DG	BW	DW	DG
155	79	266	73	60	367	47	306	147	112	127	261	48	295	157
109	76	315	54	79	367	30	300	170	28	185	287	111	110	279
174	107	219	123	63	314	42	341	117	60	174	266	144	190	166
122	80	298	151	30	319	81	141	278	105	106	289	46	262	192
346	20	134	198	39	263	55	298	147	311	33	156	146	182	172

The possible tactical line-up pairs computed by the “Public Resource Football Computing for FerSML”, v0.0.7, where "BW" means FC Basel wins, DW - Debreceni VSC wins, DG - Drawn game.

A 4-3-2-1 formációt választjuk további elemzésre.

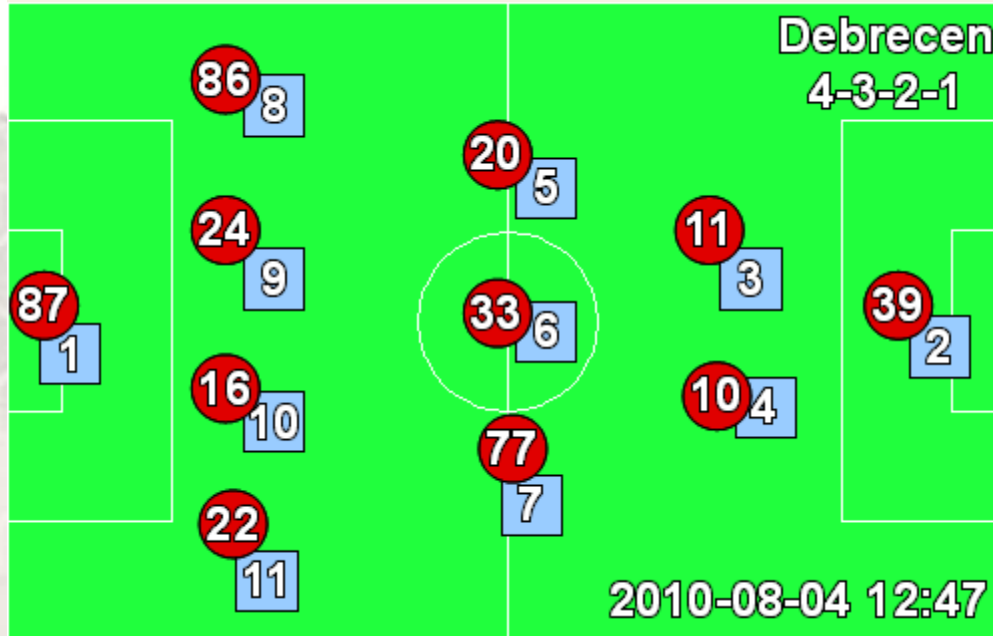
Hasonló számítások: http://fersml.blog.hu/2010/08/01/a_magyarok_labdaitol_avagy_a_bazel_debrecen_elott

N. Bátfai, E. Bátfai (2010), Public Resource Computing in European Football (submitted)

FerSML szimuláció

egy használati eset

12 órával a meccs előtt tudott kezdővel számolva



20.000 mérkőzésből

Debrecen győz ~53%

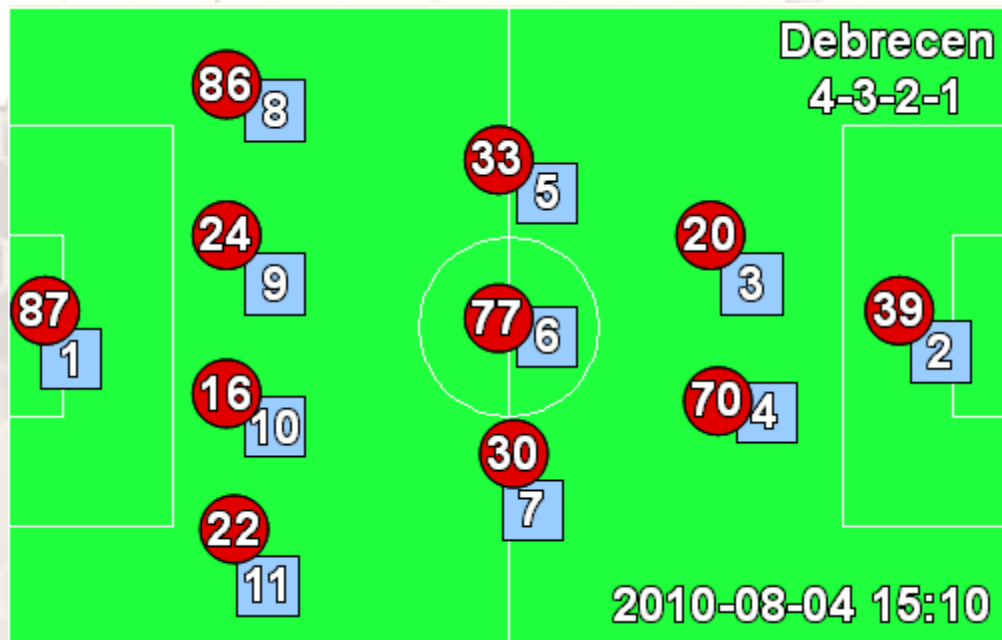
döntetlen ~42%

és csak ~5% baseli győzelem.

FerSML szimuláció

egy használati eset

néhány órával a meccs előtt tudott, a valódi kezdő tagokkal



20.000 mérkőzésből

Debrecen győz ~50 % (53)

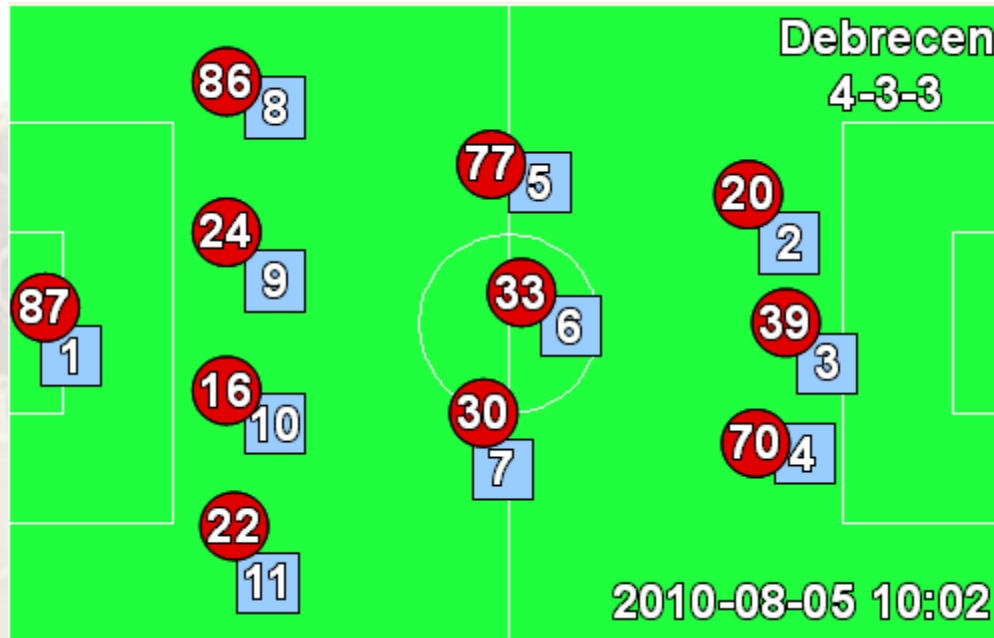
döntetlen ~47 % (42)

és csak ~13 % (5) baseli győzelem.

FerSML szimuláció

egy használati eset

de 4-3-3-at játszottunk (itt már utólag számolva)



20.000 mérkőzésből

Debrecen győz 11 % (40 ,53)

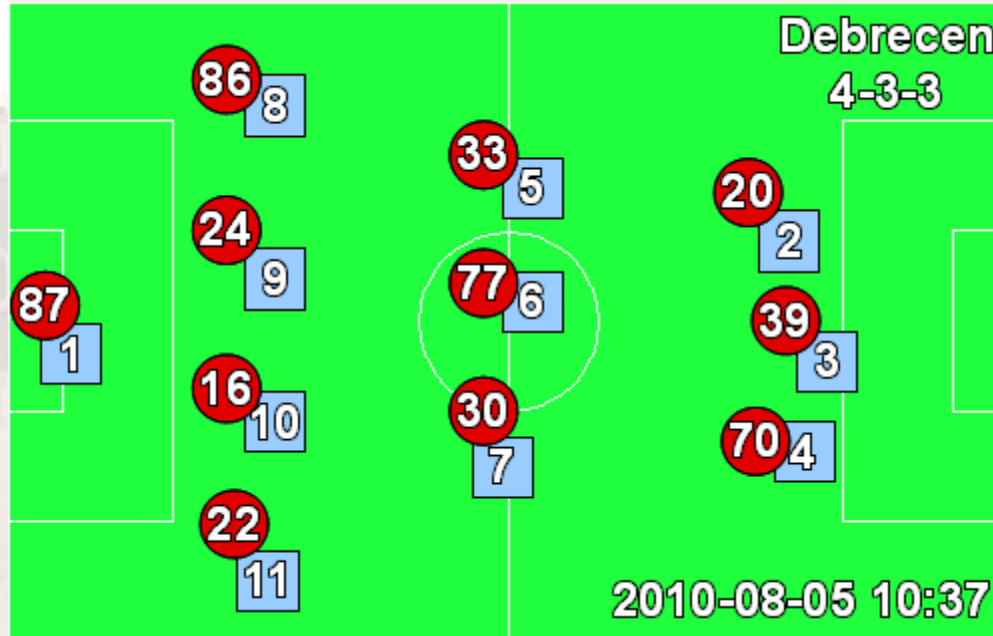
döntetlen 80 % (47 , 42)

és csak 8 % (13, 5) baseli győzelem.

FerSML szimuláció

egy használati eset

A valódi felállás:



20.000 mérkőzésből

Debrecen győz 9 % (11, 40 ,53)

döntetlen 76 % (80, 47 , 42)

és csak 15 % (8, 13, 5) baseli győzelem.

FerSML szimuláció

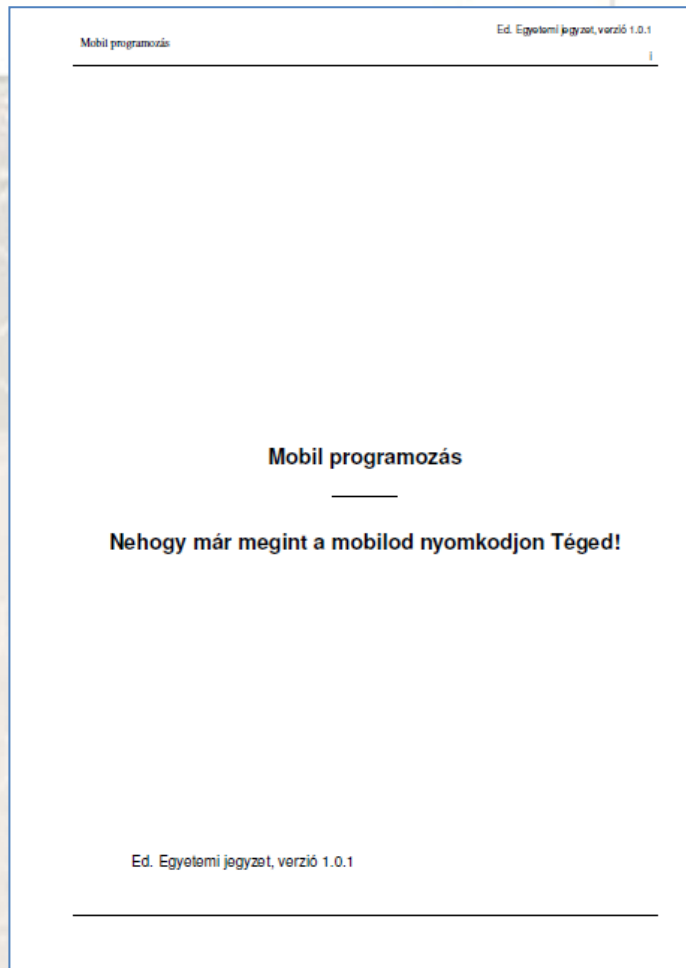
egy használati eset

*„Úgy érzem, nem érdemeltünk vereséget ezen a mérkőzésen, mert több helyzetünk volt. „ Herczeg András: Több volt ebben a mérkőzésben,
http://www.dvsc.hu/Lapok/hirek_hud.aspx?id=4187*

- i. http://fersml.blog.hu/2010/07/29/mi_tortent_az_ejjel_avagy_a_hogyan_lehetsz_avatar_fejleszto_sorozatunk_negyedik_resze
- ii. http://fersml.blog.hu/2010/07/30/a_magyarok_labdaitol_ments_meg_uram_minket
- iii. http://fersml.blog.hu/2010/08/04/lecsap_ma_bazelben_a_magyar_turul
- iv. http://fersml.blog.hu/2010/08/01/a_magyarok_labdaitol_avagy_a_bazel_debrecen_elott

A FerSML szimul

Bátfai Norbert: *Nehogy már megint*



4.1.3.1. Szurkolói avatárok

Az Automated Soccer Applet alkalmazással nem állt meg a focis játék továbbfejlesztése, jelen pillanatban a FerSML platform még mindig ezeken a kódokon alapul, az aktuális változat a Public Resource Football Computing. Ez már arra is lehetőséget ad, hogy a szurkolók által megadott XML állományok (avatárok) hajtsák meg a szimulációkat. Ez ugyanúgy Maven forrás csomagban érhető el, mint a Jávácska ONE mobil játéka. Az előző pontban említett osztályok még mindig hangsúlyosak, szerepük változatlan.

```
65
66 public interface ViewableField {
67
68     void draw();
69     void commentator(String msg);
70     void setSquadColours(int home, int away);
71     void setTeamLogo(int teamIdx, java.awt.Image logo);
72
73 }
74
```

A módszerek listája két új metódussal bővült a Automated Soccer Applet alkalmazáshoz képest. A `setSquadColours` metódus is a mobil játék ihlette, egészen pontosan ugyanazt a funkciót valósítja meg, ami a mobilos játékban is megvolt, színezi a csapatok mezét.

```
75
76 public class FootballMatch implements Runnable, FootballField, Text {
77
78     protected ViewableField display = null;
79
80 }
```

Időközben a `SFApplet` osztály elvesztette indító szerepét, illetve további interfészt, a `SimulationType` interfészt is megvalósít. Ez a szimulációk típusainak megfelelő konstans nevek használatát teszi lehetővé az osztályban.

```
81
82 public class SFApplet extends java.applet.Applet
83     implements java.awt.event.KeyListener, FootballField, ViewableField, SimulationType {
84
85     /** A Football Match that must display in this Applet. */
86     FootballMatch footballMatch;
87 }
```

Az indítás ebben a verzióban a `JSFA` swings osztályra hárul.

Szurkolói avatárok validálása GNU/Linux alatt

1. Első lépés egy szurkolói avatár elkészítése, az egyszerűség kedvéért most letöltünk egyet. Például a Debreceni VSC 2010/2011 Bajnokok Ligája selejtezőkbeli Basel FC ellen vívott márközése alapján készített: http://footballerml.sourceforge.net/supporter_avatars/0.0.9/FerSML.Debrecen.avatar.xml

```
$ wget http://footballerml.sourceforge.net/supporter_avatars/0.0.9/FerSML.Debrecen.
avatar.xml
```

2. A letöltött XML állománynak egy Relax NG kompakt szintaxissal megfogalmazott nyelvtan szerint kell érvényesnek lennie. Az ilyen nyelvtanok alapján történő validálást a `jing-trang` 20091111 verziójával tudjuk elvégezni. Innen töltsük le: <http://jing-trang.googlecode.com/files/jing-20091111.zip> Az alábbi két paranccsal letöltjük és kitöröljük a jing csomagot.

```
$ wget http://jing-trang.googlecode.com/files/jing-20091111.zip
$ unzip jing-20091111.zip
```

FerSML irodalomkutatás és célkitűzés

Sport Science Journals:

Journal of Human Sport and Exercise
Journal of Quantitative Analysis in Sports
Magyar Sporttudományi Szemle

Mesterséges
intelligencia

Sporttudomány
és „coaching”

Robot foci
(2D szimulációs liga)

A FerSML platform
az általunk fejlesztendő terület

Robotfoci vs. FerSML

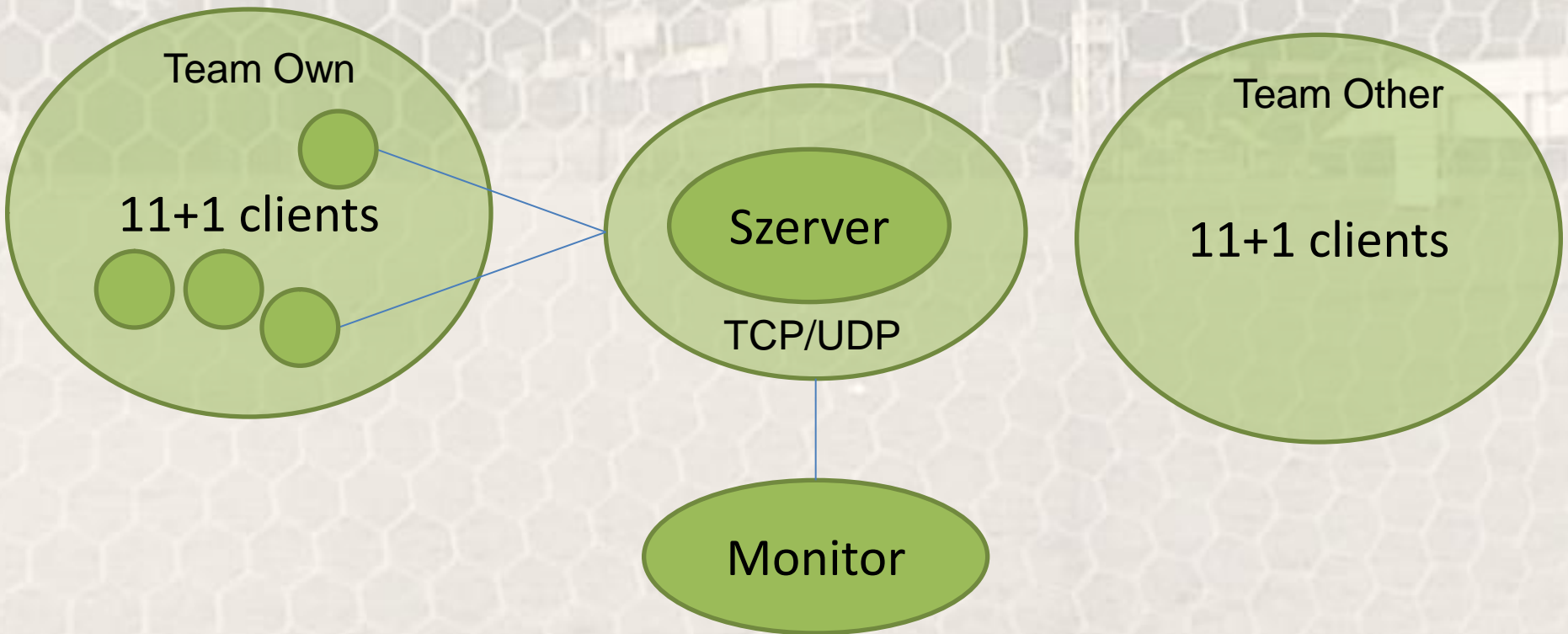
Miért alkalmatlan a RC a mi (FerSML sporttudományi) céljainkra?

Jóval magasabb absztrakciós szinten mozgunk: például a piramis alapú üzem nálunk egy belépési pont, amit a robot focinál kialakítani már komoly eredmény. (Mi nem akarunk a semmiből egy olyan játékos ágenst kialakítani, aki rendelkezik a pálya és a játék egy belső reprezentációjával, így képes intelligens viselkedésre, mert triviálisan feltesszük, hogy ez adott.)

(Aki elkészíti saját robotfocis csapatát, tapasztalni fogja, hogy ugyanaz a fejlesztői élmény, mintha csak egy LEGO robotot programozott volna.)

RoboCup Soccer 2D Simulation League

Alapcikk: Hiroaki Kitano, Minoru Asada, Yasuo Kuniyoshi, Itsuki Noda, and Eiichi Osawa. 1997. RoboCup: *The Robot World Cup Initiative*. In Proceedings of the first international conference on Autonomous agents (AGENTS '97). ACM, New York, NY, USA, 340-347.



The RoboCup Soccer Simulator

rcssserver

<https://sourceforge.net/projects/sserver/>

Egy szimulációs lépés:



6000 lépés 10 percben.

A robotfoc

Bátfai Norbert: *Mesterséges intellige* *robotfoci programozásba*

Mesterséges intelligencia a gyakorlatban: bevezetés a robotfoci programozásba

3.2.1. Programozási feladatok

Adott területen az első programozási feladatoknál sosem árthat a sok segítség, sőt! Ezért esetenként a következő feladat csokorban megadunk majd konkrét megoldásokat is.

Aranylabor FC

Készíts egy olyan Atan alapú programot, ami az Aranycsapat felállítását veszi fel a középkorokéskor!

Ha még nem tetted meg, most telepítsd az **Atan** csomagot a gépedre.

1. Vegyük azt a hadrendet, amiben a magyar nemzeti 11 felállt a híres „hat-hármas”, sokszor az évszázad mérkőzésének titulált **találkozó**n a Wembley Stadionban 1953-ban. A felállítását például ebben a **[DEIKFOCI]** cikkünkben is megtalálhatjátok.
2. Használható a következő feladat **CsakALabdaFC** és **JatekosAdapter** osztályát.
3. Nincs más dolgod, mint a középkorokéskori felállást megadni, azaz hívni a megfelelő aktuális paraméter értékkel a **move** metódust a fent mutatott kódrészletek **JatekosAdapter** osztályában.

Csak a foci FC

Készíts egy olyan Atan alapú programot, amiben a csapat minden játékosa egy emberként rohan a labda után a mérkőzés során.

1. Egy megoldási javaslatként felélelheted a **CsakALabdaFC** és a **JatekosAdapter** osztályok forrásait, létrehozhatod egy NetBeans projektet, de egyszerűbb, ha parancssorban dolgozol az alábbiak szerint:

```
C:\Users\Norbi> javac -cp Downloads\atan_0.4.3\atan.jar;Downloads\atan_0.4.3\log4j ←  
-1.2.16.jar;. CsakALabdaFC.java  
  
C:\Users\Norbi> java -cp Downloads\atan_0.4.3\atan.jar;Downloads\atan_0.4.3\log4j ←  
-1.2.16.jar;. CsakALabdaFC
```

2. Íme a **CsakALabdaFC** osztály kódja. A kód szervezéséről többet olvashatsz a **A CsordaFC osztály** pontban.

```
1 public class CsakALabdaFC extends atan.model.AbstractTeam {  
2  
3     public CsakALabdaFC(String team, int port, String host) {  
4  
5         super(team, port, host);  
6  
7     }  
8  
9     @Override  
10    public atan.model.ControllerPlayer getNewController(int number) {  
11  
12        return new JatekosAdapter();  
13    }  
14  
15  
16    public static void main(String[] args) {  
17  
18        org.apache.log4j.BasicConfigurator.configure();  
19  
20        if (args.length == 1) {  
21            new CsakALabdaFC(args[0], 6000, "localhost").connectAll();  
22        } else {  
23            new CsakALabdaFC("CsakALabdaFC", 6000, "localhost").connectAll();  
24        }  
25    }  
26  
27 }
```


Atan - atan.model.ControllerPlayer

<<interface>>

+ ControllerPlayer

**+preInfo()
+postInfo()
+getPlayer()
+setPlayer()
+getType()
+setType()
+infoSeeFlagRight()
+infoSeeFlagLeft()
+infoSeeFlagOwn()
+infoSeeFlagOther()
+infoSeeFlagCenter()
+infoSeeFlagCornerOwn()
+infoSeeFlagCornerOther()
+infoSeeFlagPenaltyOwn()
+infoSeeFlagPenaltyOther()
+infoSeeFlagGoalOwn()
+infoSeeFlagGoalOther()
+infoSeeLine()
+infoSeePlayerOther()
+infoSeePlayerOwn()
+infoSeeBall()
+infoHearReferee()
+infoHearPlayMode()
+infoHearPlayer()
+infoHearError()
+infoHearOk()
+infoHearWarning()
+infoSenseBody()**

Atan - atan.model.ControllerPlayer

```
package atan.model;
```

```
//~--- non-JDK imports ~---
```

```
import  
import  
import  
import  
import  
import  
import  
import
```

```
/**  
 * Int  
 * are  
 * is  
 * are  
 * pos  
 *  
 * @au  
 */  
public  
  
/*  
 *  
 *  
 *  
 *  
 pu
```



Home Details Sample > JavaDoc Contact Us

All Classes

Packages

[atan.model](#)
[atan.model.enums](#)
[atan.parser](#)
[atan.test](#)
[sample](#)

[atan.model](#)

Interfaces

[ActionsPlayer](#)
[ControllerPlayer](#)

Classes

[AbstractTeam](#)
[AbstractUDPClient](#)
[ByteBuffer](#)
[CommandFactory](#)
[NullLogger](#)
[SServerPlayer](#)

[Overview](#) [Package](#) **[Class](#)** [Use Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#)

DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

atan.model

Interface ControllerPlayer

All Known Implementing Classes:

[Silly](#), [Simple](#), [TestController](#)

```
public interface ControllerPlayer
```

Interface that has to be implemented in order to control players. The methods are run in a cycle whenever a see command arrives from SServer. At first preInfo() is invoked. then the info*() methods are called according to what kind of objects are currently seen or what other commands where received from the server. At last postInfo() is called. All objects are relative to the current side of the controller.

Author:

Atan

<http://atan1.sourceforge.net/javadoc.html>

Method Summary

```
public class JatekosAdapter implements atan.model.ControllerPlayer {
```

```
    private atan.model.ActionsPlayer player;
```

```
    protected boolean kozepkezdes;
```

```
    protected boolean latomAFocit;
```

```
    protected double distanceFoci;
```

```
    protected double directionFoci;
```

```
    protected double distChangeFoci;
```

```
    protected double dirChangeFoci;
```

```
    ...
```

```
    @Override
```

```
    public void preInfo() {
```

```
        kozepkezdes = false;
```

```
        latomAFocit = false;
```

```
    }
```

```
    ...
```

```
    @Override
```

```
    public void infoSeeBall(double distance, double direction, double d:  
        double bodyFacingDirection, double headFacingDirection) {
```

```
        latomAFocit = true;
```

```
        distanceFoci = distance;
```

```
        directionFoci = direction;
```

```
        dirChangeFoci = distChange;
```

```
        dirChangeFoci = dirChange;
```

```
    }
```

```
<<interface>>
```

```
+ ControllerPla
```

```
+preInfo()
```

```
+postInfo()
```

```
+getPlayer()
```

```
+setPlayer()
```

```
+getType()
```

```
+setType()
```

```
+infoSeeFlagRight()
```

```
+infoSeeFlagLeft()
```

```
+infoSeeFlagOwn()
```

```
+infoSeeFlagOther()
```

```
+infoSeeFlagCenter()
```

```
+infoSeeFlagCornerO
```

```
+infoSeeFlagPenaltyC
```

```
+infoSeeFlagPenaltyC
```

```
+infoSeeFlagGoalOwi
```

```
+infoSeeFlagGoalOth
```

```
+infoSeeLine()
```

```
+infoSeePlayerOtherC
```

```
+infoSeePlayerOwn()
```

```
+infoSeeBall()
```

```
+infoHearReferee()
```

```
+infoHearPlayMode()
```

```
+infoHearPlayer()
```

```
+infoHearError()
```

```
+infoHearOk()
```

```
+infoHearWarning()
```

```
+infoSenseBody()
```

Atan - atan.model.ActionsPlayer

<<interface>>

+ ActionsPlayer

+dash()

```
package atan.model;

//~--- non-JDK imports -----

import atan.model.enums.ViewAngle;
import atan.model.enums.ViewQuality;

/**
 * Interface for an abstract soccer player. To be used by ControllerPlayer.
 */
public interface ActionsPlayer {

    /**
     * This command accelerates the player in the direction of its body.
     * @param power Power is between minpower (-100) and maxpower (+100).
     */
    public void dash(int power);

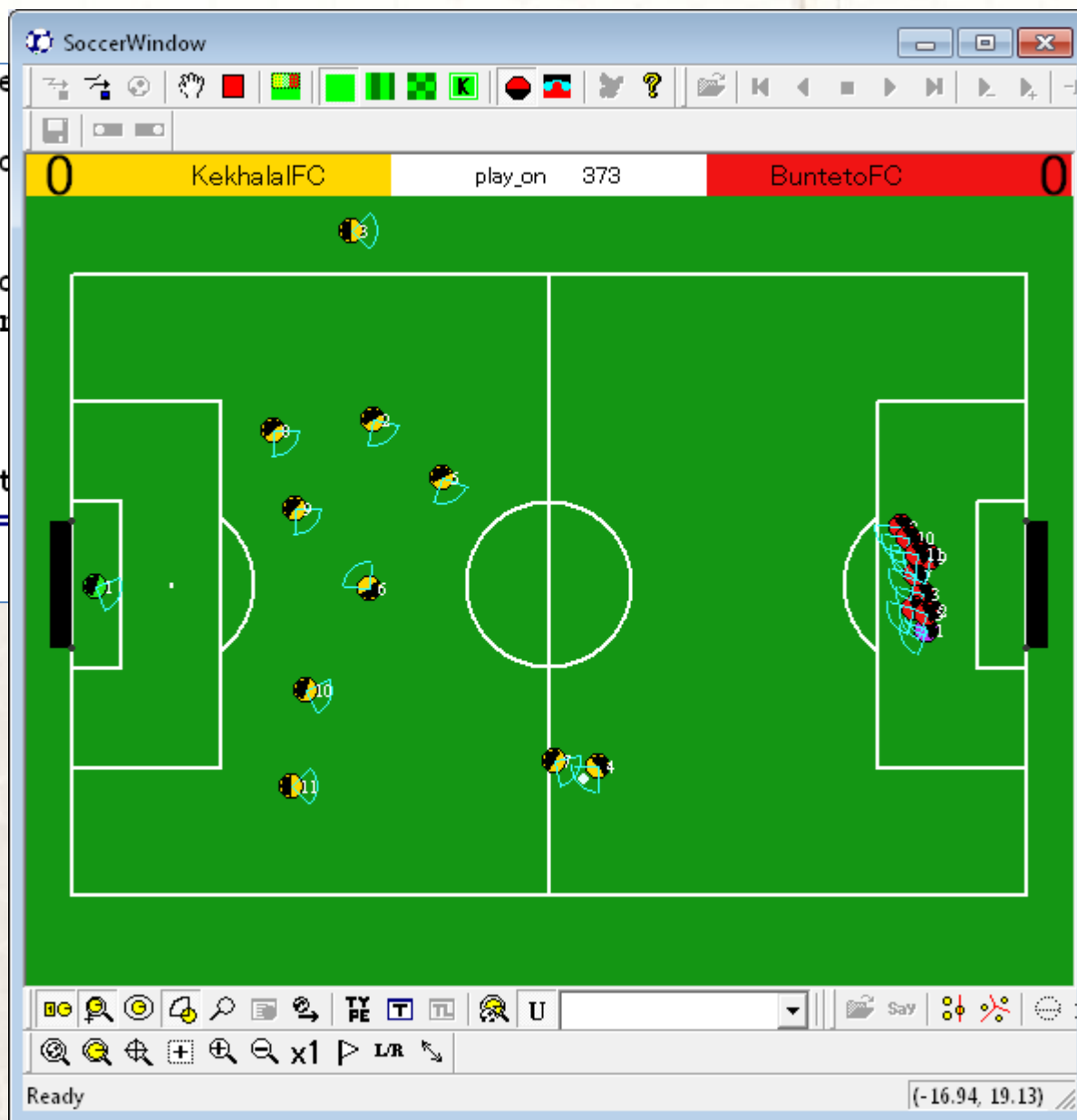
    /**
     * This command can only be executed before kick off or after a goal.
     * @param x X location (between -54 and +54).
     * @param y Y location (between -32 and +32).
     */
    public void move(int x, int y);
}
```

Büntető FC

```
public class BuntetoFC extends atan.model.AbstractTeam {  
  
    public BuntetoFC(String team, int port, String host) {  
        super(team, port, host);  
    }  
  
    @Override  
    public atan.model.ControllerPlayer getNewController(int number) {  
        return new JatekosAdapter();  
    }  
  
    public static void main(String[] args) {  
        org.apache.log4j.BasicConfigurator.configure();  
  
        if (args.length == 1) {  
            new BuntetoFC(args[0], 6000, "localhost").connectAll();  
        } else {  
            new BuntetoFC("BuntetoFC", 6000, "localhost").connectAll();  
        }  
    }  
}
```

Büntető FC

```
public class Jatek {  
  
    private atan.mod  
  
    @Override  
    public atan.mod  
        return player  
    }  
  
    @Override  
    public void set  
        this.player =  
    }  
}
```



```
per());
```

Ready

(-16.94, 19.13)

BNF, Backus Normal Form (P1 ism.)

John Backus, ALGOL 60
Környezetfüggetlen nyelvekhez

<nem terminális> ::= konkatenációja terminálisoknak, nem terminálisoknak, illetve {iteráció}, [opcionális], alter|natíva

<egész szám> ::= <előjel><szám>

<előjel> ::= [-|+]

<szám> ::= <számjegy>{<számjegy>}

<számjegy> ::= 0|1|2|3|4|5|6|7|8|9

RCSS protokollok

A kliens ágensek érzékelési protokollja

```
ObjInfo ::= (ObjName Distance Direction DistChange DirChange BodyFacingDir HeadFacingDir )
           | (ObjName Distance Direction DistChange DirChange
           | (ObjName Distance Direction)
           | (ObjName Direction)
ObjName ::= (p ["Teamname" [UniformNumber [goalie]]])
           | (b)
           | (g [l|r])
           | (f c)
           | (f [l|c|r] [t|b])
           | (f p [l|r] [t|c|b])
           | (f g [l|r] [t|b])
           | (f [l|r|t|b] 0)
           | (f [t|b] [l|r] [10|20|30|40|50])
           | (f [l|r] [t|b] [10|20|30])
           | (l [l|r|t|b])
           | (B)
           | (F)
           | (G)
           | (P)
```

Distance ::= positive real number

Direction ::= -180 ~180 degrees

DistChange ::= real number

DirChange ::= real number

HeadFaceDir ::= -180 ~180 degrees

BodyFaceDir ::= -180 ~180 degrees

Teamname ::= string

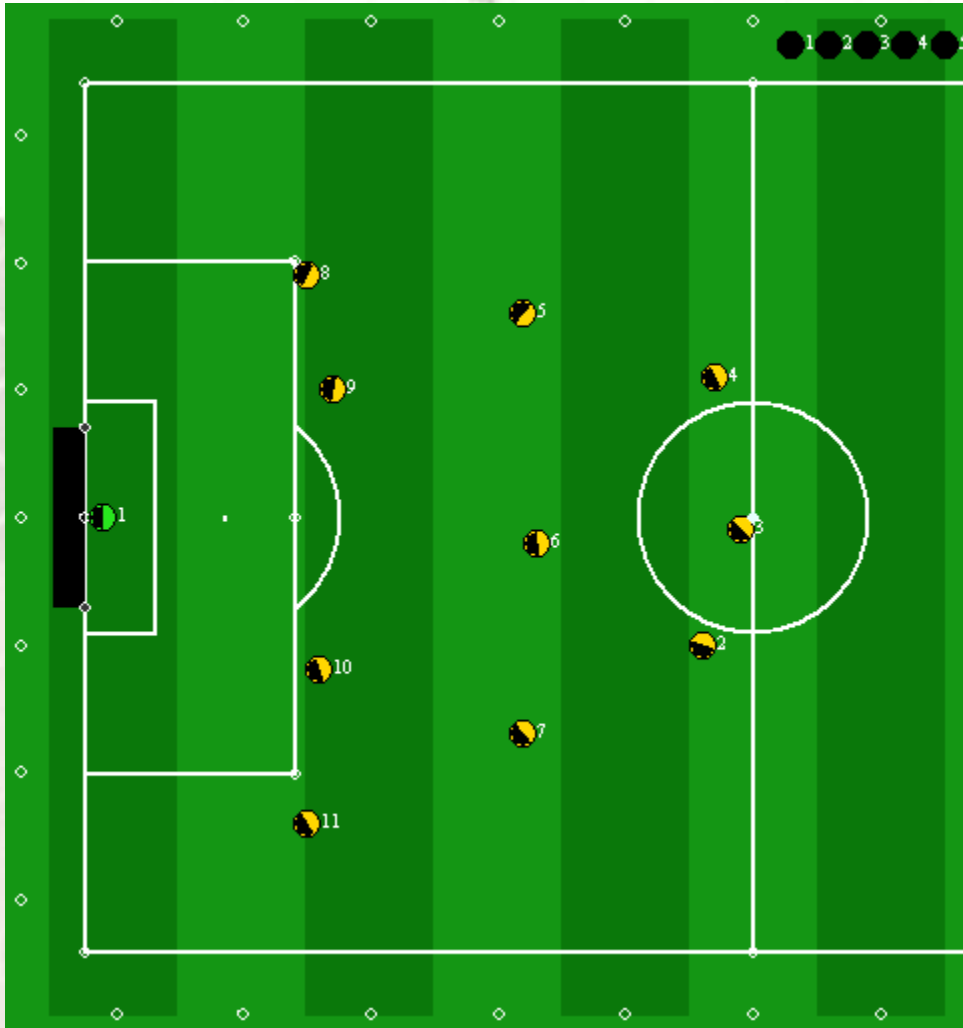
UniformNumber ::= 1 ~11

<http://netcologne.dl.sourceforge.net/project/sserver/rcssmanual/9-20030211/manual-20030211.pdf>

<http://sourceforge.net/projects/sserver/files/rcssmanual/>

RCSS protokollok

A pálya



```
/* 4-3-3 */
@Override
protected void kozepkezdes() {

    switch (getPlayer().getNumber()) {

        // Vedo

        case 8:
            getPlayer().move(-35, -19);
            break;

        case 9:
            getPlayer().move(-33, -10);
            break;

        case 10:
            getPlayer().move(-34, 12);
            break;

        case 11:
            getPlayer().move(-35, 24);
            break;

    }

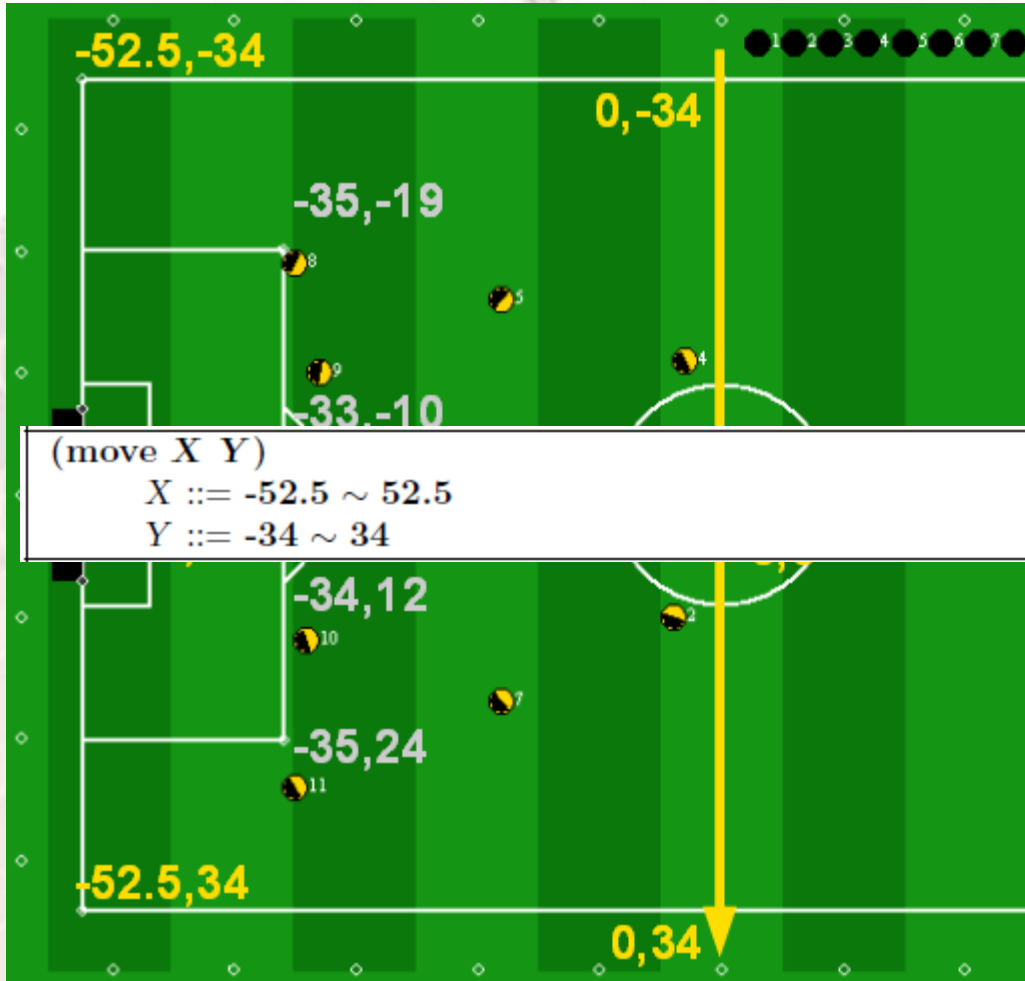
}
```

<http://netcologne.sourceforge.net/project/sserver/rcssmanual/9-20050211/manual-20050211.pdf>

<http://sourceforge.net/projects/sserver/files/rcssmanual/>

RCSS protokollok

A pálya



(move X Y)

X ::= -52.5 ~ 52.5

Y ::= -34 ~ 34

```
/* 4-3-3 */
@Override
protected void kozepkezdes() {

    switch (getPlayer().getNumber()) {

        // Vedo

        case 8:
            getPlayer().move(-35, -19);
            break;

        case 10:
            getPlayer().move(-34, 12);
            break;

        case 11:
            getPlayer().move(-35, 24);
            break;

    }

}
```

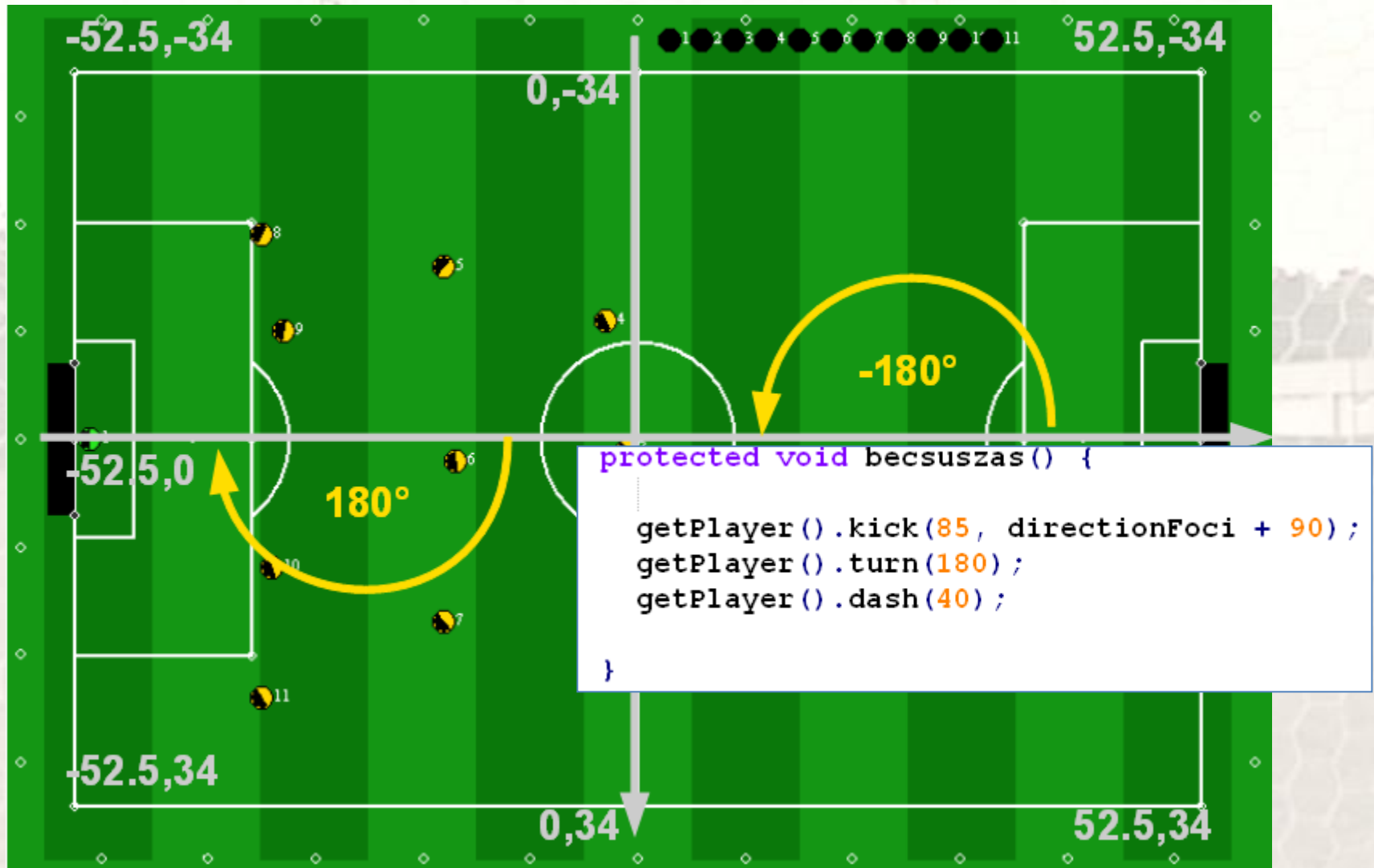
Yes

<http://netcologne.dl.sourceforge.net/project/sserver/rcssmanual/9-20050211/manual-20050211.pdf>

<http://sourceforge.net/projects/sserver/files/rcssmanual/>

RCSS protokollok

A szögek értelmezése a pályán



<http://netcologne.dl.sourceforge.net/project/sserver/rcssmanual/9-20030211/manual-20030211.pdf>

<http://sourceforge.net/projects/sserver/files/rcssmanual/>

RoboCup

Saját Atan alapú RCSS csapat fejlesztése.

Részletek majd a 4. előadásban!

Software Engineering

Szoftver fejlesztés



Szoftver krízis

1) The 1968/69 NATO Software Engineering Reports

<http://homepages.cs.ncl.ac.uk/brian.randell/NATO/NATOREports/index.html>

2) Edsger W. Dijkstra. 1972. *The humble programmer*. Commun. ACM 15, 10 (October 1972), 859-866. DOI=10.1145/355604.361591

<http://doi.acm.org/10.1145/355604.361591>,

pdf-ben az egyetemi gépekről:

<http://dl.acm.org/citation.cfm?id=355604.361591&coll=DL&dl=ACM&CFID=48433466&CFTOKEN=41884745>

születik egy szakma

retical physicist, or to carry my study of physics to a formal completion only, with a minimum of effort, and to become . . . , yes what? A programmer? But was that a respectable profession? After all, what was programming? Where was the sound

Megoldás: technológizáljuk a szoftverfejlesztés folyamatát, legyen egy mérnöki tudomány.

A szoftver fejlesztés modellezése

Szoftverfolyamat: **specifikáció, tervezés, fejlesztés, validálás, evolúció**; ezeket kombinálják, permutálják, iterálják elképzeléseik szerint a következő modellek

- 1) Vízesés (waterfall model)
- 2) Evolúciós (software prototyping)
- 3) Spirális (spiral model)
- 4) Komponens alapú (Commercial Off The Shelf = COTS , COTS-based software development)
- 5) Agilis (agile software development)

A szoftver fejlesztés modellezése

Eric S. Raymond: A katedrális és a bazar

<http://magyar-irodalom.elte.hu/robert/szovegek/bazar/>

RERO: Release Early, Release Often

„Cowboy coding”

A programozás eszközei

- 1) IDE
- 2) Verzió kezelő (cvs, svn)
- 3) Build eszköz (make, Ant, Maven)
Maven + tesztelés, dokumentálás

A Maven projektek tárgya (artifact)

A Jávácska ONE/FocijatekNekedOSE projekt tárgya: **focijatek-neked-1.0.0-me.jar**

A PublicResourceFCforFerSML projekt tárgya: **PublicResourceFCforFerSML-0.0.16.jar**

Az AranycsapatFC projekt tárgya: **AranycsapatFC-0.0.1.jar**

Miket használtunk az előző laboron?

- atan.jar
- log4j-1.2.16.jar

A Maven központi (tárgy) tároló

The screenshot displays the Maven Central repository interface. At the top, there is a search bar with the text "Search by group, artifact or description. E.g: taglib, spring, hibernate". Below this, the breadcrumb navigation shows "home » log4j » log4j » 1.2.16". The main content area features the "Maven Central beta" logo and a search input field containing "log4j". The search results are displayed in a table with the following columns: GroupId, ArtifactId, Latest Version, Updated, and Downloads. The results show two artifacts: "log4j" with the latest version "1.2.16" and "apache-log4j-extras" with the latest version "1.1".

<http://search.maven.org/#search|ga|1|log4j>

```
<dependency>
  <groupId>log4j</groupId>
  <artifactId>log4j</artifactId>
  <version>1.2.16</version>
</dependency>
```

This artifact depends on ...

Group	Artifact	Version
javax.mail	mail	1.4.1
junit	junit	3.8.2

<http://mvnrepository.com/artifact/log4j/log4j/1.2.16>

A Maven központi (tárgy) tároló

MVN REPOSITORY

+1 367

Search by group, artifact or description.
E.g: taglib, spring, hibernate

home » log4j » log4j » 1.2.16

Repository

- Plugins
- Tag Cloud

Artifacts/Jars

249k
124k
0

2004 2011

Artifacts/Year

Feeds

RSS 2.0
ATOM FEED

Popular Tags

ajax analysis annotations ant apache api archetype aspect asynchronously beans binding bpm build buildsystem bytecode cache CMS codecoverage codehaus collections concurrency container database directory

Apache Log4j

Apache Log4j 1.2

Data Storage in the Cloud

See how Hitachi Cloud Storage can free up valuable I.T. resources.
HDS.com/Cloud

AdChoices ▶

Artifact	Download (JAR) (471 KB)
POM File	View
HomePage	http://logging.apache.org/log4j/1.2/
Organization	Apache Software Foundation
Issue Tracker	http://issues.apache.org/bugzilla/

Apache Maven Apache Ivy (Ant) Grape

```
<dependency>
  <groupId>log4j</groupId>
  <artifactId>log4j</artifactId>
  <version>1.2.16</version>
</dependency>
```

This artifact depends on ...

Group	Artifact	Version
javax.mail	mail	1.4.1
junit	junit	3.8.2

A tárgy Maven koordinátái

```
<dependency>
  <groupId>log4j</groupId>
  <artifactId>log4j</artifactId>
  <version>1.2.16</version>
</dependency>
```

Maven koordináták:

1. csoport azonosító
2. tárgy azonosító
3. verziószám

Függőségek deklarációja

```
<dependencies>
  <dependency>
    <groupId>maven</groupId>
    <artifactId>maven-jar-plugin</artifactId>
    <version>2.3.2</version>
  </dependency>
  <dependency>
    <groupId>log4j</groupId>
    <artifactId>log4j</artifactId>
    <version>1.2.16</version>
  </dependency>
  <dependency>
    <groupId>atan</groupId>
    <artifactId>atan</artifactId>
    <version>0.4.3</version>
  </dependency>
</dependencies>
```

Projektünkben függőségként szerepel

AranycsapatFC / pom.xml

Tárgy beillesztése a lokális repóba

```
Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Norbi>cd Documents\MavenProjects\AranycsapatFC

C:\Users\Norbi\Documents\MavenProjects\AranycsapatFC>mvn install:install-file -D
file=c:/Users/Norbi/Downloads/atan_0.4.3/atan.jar -DgroupId=atan -DartifactId=at
an -Dversion=0.4.3 -Dpackaging=jar
```

```
Command Prompt
C:\Users\Norbi>cd Documents\MavenProjects\AranycsapatFC

C:\Users\Norbi\Documents\MavenProjects\AranycsapatFC>mvn install:install-file -D
file=c:/Users/Norbi/Downloads/atan_0.4.3/atan.jar -DgroupId=atan -DartifactId=at
an -Dversion=0.4.3 -Dpackaging=jar
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building AranycsapatFC 0.0.1
[INFO] -----
[INFO]
[INFO] --- maven-install-plugin:2.3.1:install-file (default-cli) @ AranycsapatFC
---
[INFO] Installing c:\Users\Norbi\Downloads\atan_0.4.3\atan.jar to C:\Users\Norbi
\.m2\repository\atan\atan\0.4.3\atan-0.4.3.jar
[INFO] Installing C:\Users\Norbi\AppData\Local\Temp\mvninstall1839598219902766633
1.pom to C:\Users\Norbi\.m2\repository\atan\atan\0.4.3\atan-0.4.3.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.046s
[INFO] Finished at: Sun Sep 25 14:42:41 CEST 2011
[INFO] Final Memory: 3M/15M
[INFO] -----
C:\Users\Norbi\Documents\MavenProjects\AranycsapatFC>
```

POM (Projekt Objektum Modell)

pom.xml

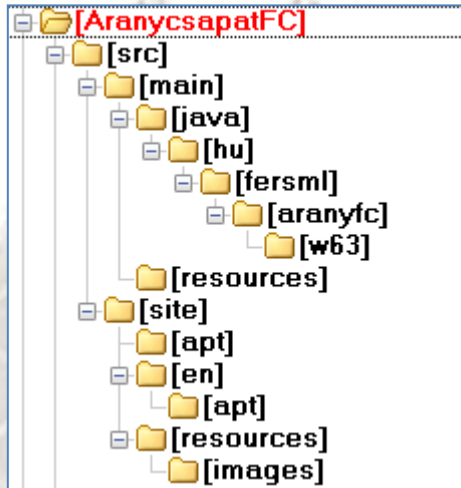
```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v
<modelVersion>4.0.0</modelVersion>
<groupId>hu.fersml.aranyfc</groupId>
<artifactId>AranycsapatFC</artifactId>
<packaging>jar</packaging>
<version>0.0.1</version>
<name>AranycsapatFC</name>
<url>http://progpater.blog.hu/2011/09/23/we_re_a_unit_not_a_one_man_show</url>
<description>A "Mesterséges intelligencia a gyakorlatban: bevezetés a robotfoci programozásb
<organization>
  <name>Batfai Norbert, University of Debrecen, Department of Information Technology</name>
  <url>http://www.inf.unideb.hu/~nbatfai</url>
</organization>
<licenses>
  <license>
    <name>GNU GPL v3</name>
    <url>http://www.gnu.org/licenses/gpl.html</url>
  </license>
</licenses>
<developers>
  <developer>
    <id>norbi</id>
    <name>Norbert Batfai</name>
    <email>nbatfai@gmail.com</email>
```

POM (Projekt Objektum Modell)

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>hu.ferret</groupId>
  <artifactId>Arany</artifactId>
  <packaging>jar</packaging>
  <version>0.0.1</version>
  <name>Aranycsapat</name>
  <url>http://prog.hu</url>
  <description>A "M...</description>
  <organization>
    <name>Batfai No...</name>
    <url>http://www...</url>
  </organization>
  <licenses>
    <license>
      <name>GNU GPL</name>
      <url>http://w...</url>
    </license>
  </licenses>
  <developers>
    <developer>
      <id>norbi</id>
      <name>Norbert</name>
      <email>nbatfa...</email>
    </developer>
  </developers>
  <dependencies>
    <dependency>
      <groupId>maven</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <version>2.3.2</version>
    </dependency>
    <dependency>
      <groupId>log4j</groupId>
      <artifactId>log4j</artifactId>
      <version>1.2.16</version>
    </dependency>
    <dependency>
      <groupId>atan</groupId>
      <artifactId>atan</artifactId>
      <version>0.4.3</version>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
      </plugin>
    </plugins>
  </build>
</project>
```

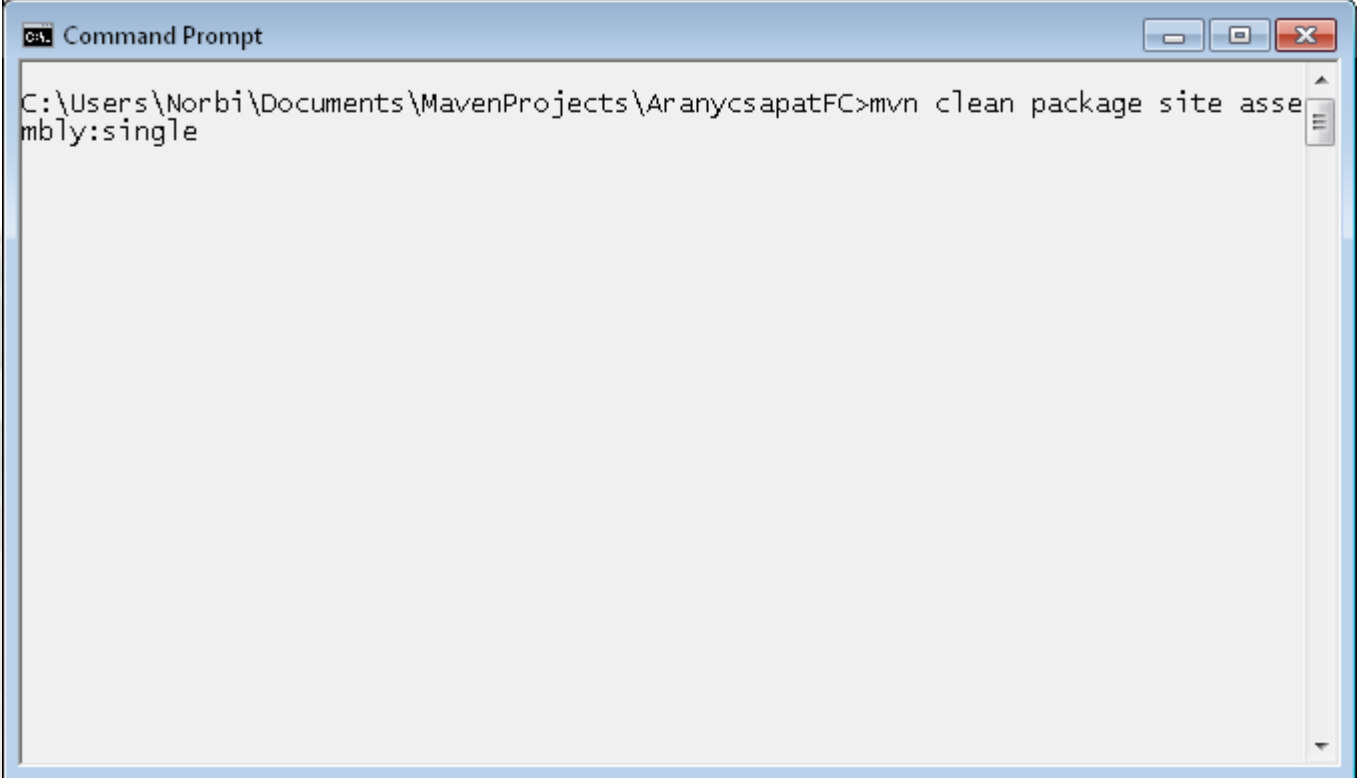

A projekt szervezése



Az src könyvtárral egy szinten van a pom.xml

Életciklus

Moduláris felépítés:
bővítmények, célok

A screenshot of a Windows Command Prompt window. The title bar reads "Command Prompt". The command prompt shows the current directory as "C:\Users\Norbi\Documents\MavenProjects\AranycsapatFC" and the command being executed is "mvn clean package site assembly:single".

```
Command Prompt
C:\Users\Norbi\Documents\MavenProjects\AranycsapatFC>mvn clean package site assembly:single
```

Életciklus, a projekt fejlesztői szájtja

AranycsapatFC

Last Published: 2011-09-25 | Version: 0.0.1

Magyar

- Download
- Project Documentation
 - Project Information
 - Project Reports
 - Checkstyle
 - Cobertura Test
 - Coverage
 - CPD Report
 - JavaNCSS Report
 - JDepend
 - PMD Report
 - Source Xref
 - Surefire Report



Checkstyle Results

The following document contains the results of Checkstyle

Summary

Files	Infos	Warnings	Errors
18	0	0	810

Files

Files	I	W	E
hu/fersml/aranyfc/AranycsapatFC.java	0	0	23
hu/fersml/aranyfc/Fedezet.java	0	0	65
hu/fersml/aranyfc/Jatekos.java	0	0	301
hu/fersml/aranyfc/JatekosAdapter.java	0	0	208
hu/fersml/aranyfc/Kapus.java	0	0	36
hu/fersml/aranyfc/Tamado.java	0	0	13
hu/fersml/aranyfc/Vedo.java	0	0	79
hu/fersml/aranyfc/w63/Bozsik.java	0	0	8
hu/fersml/aranyfc/w63/Budai.java	0	0	9
hu/fersml/aranyfc/w63/Buzanszky.java	0	0	9
hu/fersml/aranyfc/w63/Czibor.java	0	0	7
hu/fersml/aranyfc/w63/Grosics.java	0	0	6
hu/fersml/aranyfc/w63/Hidegkuti.java	0	0	6
hu/fersml/aranyfc/w63/Kocsis.java	0	0	9
hu/fersml/aranyfc/w63/Lantos.java	0	0	9
hu/fersml/aranyfc/w63/Lorant.java	0	0	6
hu/fersml/aranyfc/w63/Ruskas.java	0	0	8

erved.

Életciklus, a projekt terjesztése

AranycsapatFC

Last Published: 2011-09-25 | Version: 0.0.1

Magyar

Download

Download

Project

Documentation

- ▶ [Project Information](#)
- ▶ [Project Reports](#)



A AranycsapatFC letöltése

- Letöltés
 - [AranycsapatFC-0.0.1-project.zip](#)
 - [AranycsapatFC-0.0.1-project.tar.gz](#)
 - [AranycsapatFC-0.0.1-project.tar.bz2](#)

Copyright © 2011 Batfai Norbert, University of Debrecen, Department of Information Technology. All Rights Reserved.

Szoftver metrikák

- 1) Ciklomatikus komplexitás: a vezérlés bonyolultsága
- 2) OO metrikák

● SZOFTVERFEJLESZTÉS

Van-e az objektum-orientált programoknak anyanyelve? – avagy egy analitikai szöveg bevezetése

BÁTFAI NORBERT

*Debreceni Egyetem, Informatikai Kar, Információ Technológia Tanszék
batfai.norbert@inf.unideb.hu*

<mailto:batfai.norbert@inf.unideb.hu>

Kulcsszavak: szoftverfejlesztés, információ-visszakeresés, AOP mérések, AspectJ, Java, PageRank, Zipf, Ant, Maven

Rövid írásunkban Java parancssori, Ant vagy Maven forrásprojektekhez vezetünk be olyan AspectJ aspektust, ami analitikát sző a program futásába. Az osztályok és a metódusok halmazára alkalmazza a PageRank algoritmust, illetve az objektumok kommunikációjára heurisztikusan ellenőrzi a Zipf törvény teljesülését.

http://www.hiradastechnika.hu/data/upload/file/2010/HT2011_2_komplett.pdf

Ciklomatikus komplexitás

```
@Override
protected void jatekbanVezerles() {
    if (latomAKozepet && distanceKozep < 10.0) {
        sarkonFordul();
    } else if (latomASajatGolvonalat && distanceSajatGolvonal < 0.5) {
        sarkonFordul();
    } else if (latomASzelet && distanceSzele < 5.5) {
        palyanMaradni();
    } else if (latomAFocit) {
        if (distanceFoci < 0.7) {
            rugdEsFuss();
        } else if (distanceFoci < 1.4) {
            becsuszas();
        } else if (distanceFoci < 15.0) {
            egyuttElaJatekkal(100);
        } else if (distanceFoci < 25.0) {
            egyuttElaJatekkal(80);
        } else if (distanceFoci < 30.0) {
            egyuttElaJatekkal(70);
        } else if (distanceFoci < 35.0) {
            egyuttElaJatekkal(40);
        } else {
            egyuttElaJatekkal(5);
        }
    } else {
        teblabol();
    }
}
```

AranycsapatFCa043\src\main\java\hu\fersml\aranyfc\Vedo.java

Ciklomatikus komplexitás

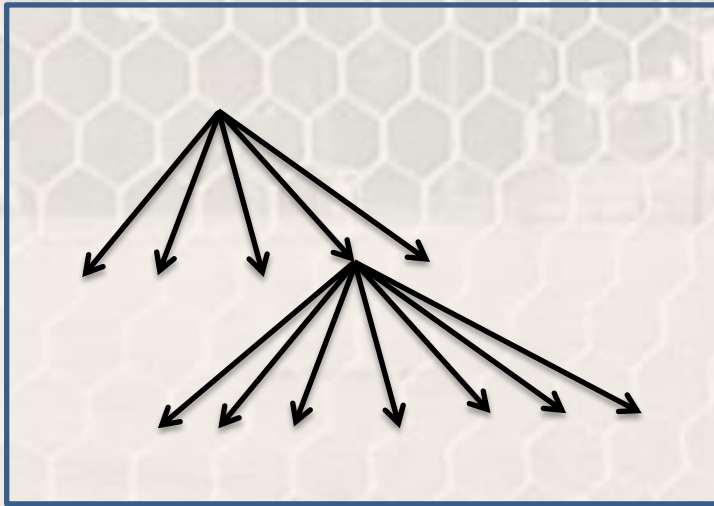
```
if (latomAKozepet && distanceKozep < 10.0)  
    sarkonFordul();
```

```
} else if (latomASajatGolvonalat && distanceSajatGolvon  
    sarkonFordul();
```

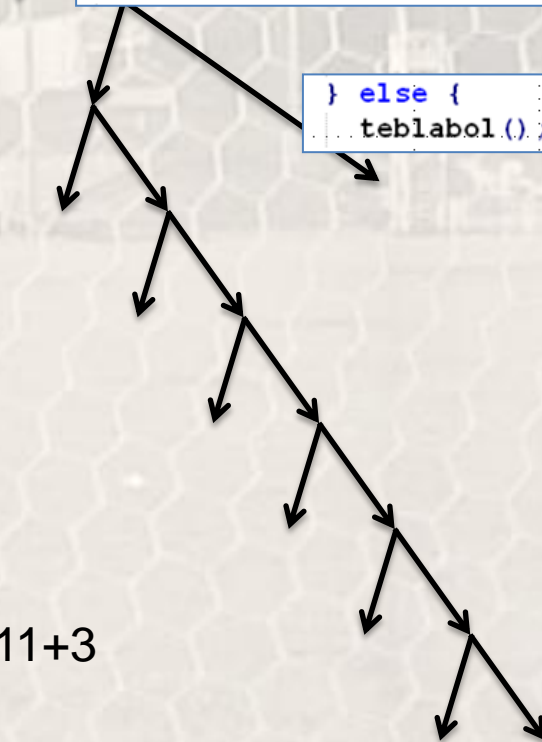
```
} else if (latomASzelet && distanceSzele < 5.5) {  
    palyanMaradni();
```

```
} else if (latomAFocit) {
```

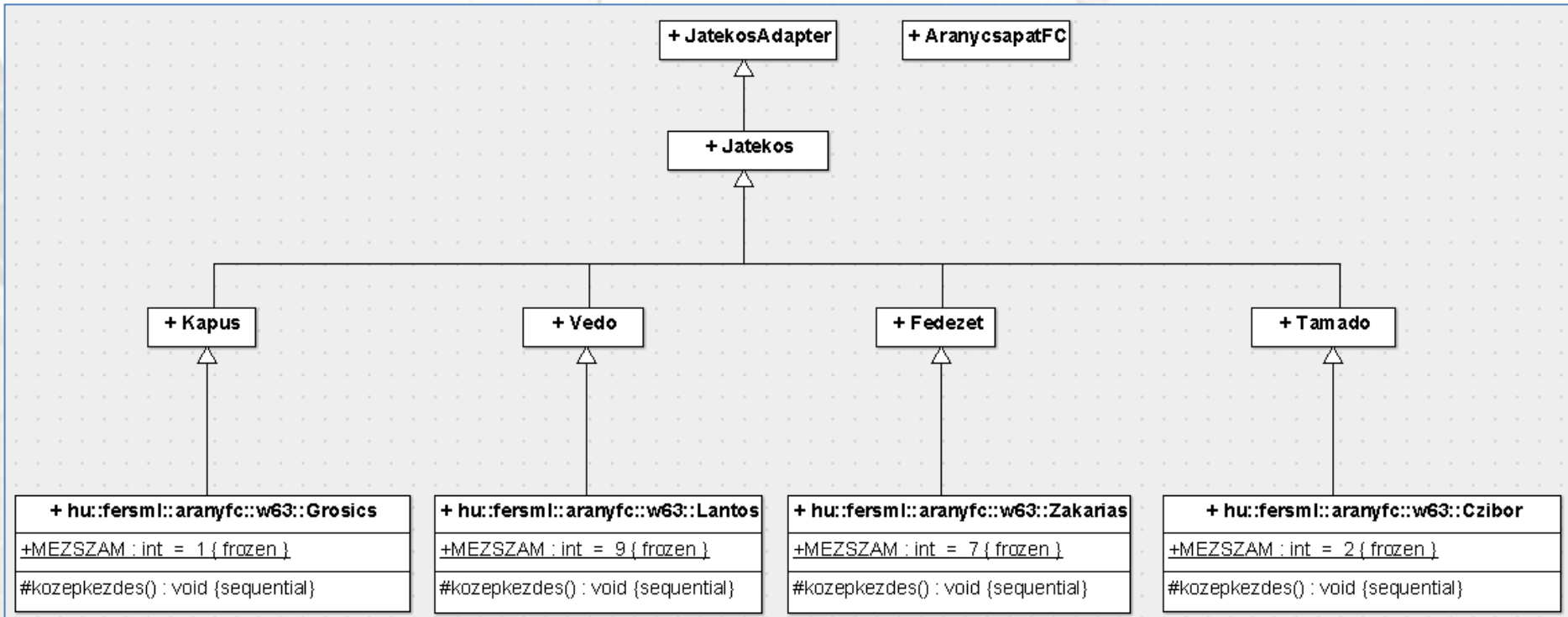
```
} else {  
    teblabol();
```



$$14 = 11 + 3$$



Aranycsapat FC



Aranycsapat FC

```
package hu.fersml.aranyfc;
```

```
public class AranycsapatFC extends atan.model.AbstractTeam {
```

```
    public AranycsapatFC(String team, int port, String host) {
```

```
        super(team, port, host);
```

```
    }
```

```
@Override
```

```
public atan.mod
```

```
    hu.fersml.ara
```

```
    switch (numbe
```

```
        case 0:
```

```
            jatekos =  
            break;
```

```
        case 10:
```

```
            jatekos =  
            break;
```

```
        case 9:
```

```
            jatekos = new hu.fersml.aranyfc.w63.Lorant();  
            break;
```

```
        case 3:
```

```
            jatekos = new hu.fersml.aranyfc.w63.Hidegkuti();  
            break;
```

```
        case 2:
```

```
            jatekos = new hu.fersml.aranyfc.w63.Puskas();  
            break;
```

```
        case 1:
```

```
            jatekos = new hu.fersml.aranyfc.w63.Czibor();  
            break;
```

```
    }
```

```
    return jatekos;
```

```
}
```

Aranycsapat FC

```
package hu.fersml.aranyfc.w63;
```

```
public class Buzanszky extends hu.fersml.aranyfc.Vedo {
```

```
    public static final int MEZSZAM = 11;
```

```
    private static org.apache.  
        org.apache.log4j.L
```

```
    @Override
```

```
    protected void kozepkezdes
```

```
        logger.info("KOZEPKEZDES  
            + getPlayer().ge  
            + " tavolsaga =  
            + " iranya = " +
```

```
        switch (getPlayer().getN
```

```
            case MEZSZAM:
```

```
                getPlayer().move(-30  
                    break;
```

```
        }
```

```
    }
```

```
}
```

```
    @Override
```

```
    public void postInfo() {
```

```
        if (latomAFocit && distanceFoci < 4.0) {  
            mozoghat = true;  
        }
```

```
        if (kirugas) {  
            kirugashoz();  
        } else if (kozepkezdes) {  
            kozepkezdes();  
        } else {
```

```
            jatekbanVezerles();  
        }
```

```
    }
```

```
    @Override
```

```
    protected void jatekbanVezerles() {
```

```
        if (latomAKozepet && distanceKozep < 10.0) {  
            sarkonFordul();
```

hu.fersml.aranyfc.Vedo

Code Conventions for the Java Programming Language

```
/*
 * Szereplő.java
 */
/**
 * Létrehoz egy paraméterben kapott szerkezetű <code>Labirintus</code>
 * objektumot.
 *
 * @param kincsekSzama a kincsek száma a labirintusban.
 * @param szörnyekSzama a szörnyek száma a labirintusban.
 * @exception RosszLabirintusKivétel ha a labirintust definiáló tömb
 * nincs elkészítve.
 */
public Labirintus(boolean[][] szerkezet, int kincsekSzama, int szörnyekSzama)
throws RosszLabirintusKivétel {

    if(szerkezet == null)
        throw new RosszLabirintusKivétel("A labirintust definiáló tömb nincs elkészítve.");

    this.szerkezet = szerkezet;

    magasság = szerkezet.length;
    szélesség = szerkezet[0].length;

    kincsekSzörnyek(kincsekSzama, szörnyekSzama);
}
```

Laborkártyák

A KeyAdapter nem valósítja meg a KeyListener ősének, az EventListener interfésznek a módszereit... magyarázd meg a helyzetet!

```
public interface KeyListener extends EventListener {  
  
    /**  
     * Invoked when a key has been typed.  
     * See the class description for {@link KeyEvent} for a definition of  
     * a key typed event.  
     */  
    public void keyTyped(KeyEvent e);  
  
    /**  
     * Invoked when a key has been pressed.  
     * See the class description for {@link KeyEvent} for a definition of  
     * a key pressed event.  
     */  
    public void keyPressed(KeyEvent e);  
  
    /**  
     * Invoked when a key has been released.  
     * See the class description for {@link KeyEvent} for a definition of  
     * a key released event.  
     */  
    public void keyReleased(KeyEvent e);  
}
```

```
public abstract class KeyAdapter implements KeyListener {  
  
    /**  
     * Invoked when a key has been typed.  
     * This event occurs when a key press is followed by a key release.  
     */  
    public void keyTyped(KeyEvent e) {}  
  
    /**  
     * Invoked when a key has been pressed.  
     */  
    public void keyPressed(KeyEvent e) {}  
  
    /**  
     * Invoked when a key has been released.  
     */  
    public void keyReleased(KeyEvent e) {}  
}
```

Laborkártyák

Sorolj fel, mutass meg 5 konkrét konvenciót, saját programodbeli példával!

Code Conventions for the Java Programming Language

<http://www.oracle.com/technetwork/java/codeconv-138413.html>

Laborkártyák

Melyik a helyes, miért?

- 1) Nem fordul le, mert hibás az előírt lépésszámú ciklus (hiányos).
- 2) 10 db „a”-t és egy „b”-t külön sorokban.
- 3) 11 db „a”-t és egy „b”-t külön sorokban.
- 4) 12 db „a”-t és egy „b”-t külön sorokban.
- 5) Nem fordul le, mert nincs goto a nyelvben.
- 6) Nem fordul le más okból.
- 7) Lefordul, ha a goto kulcsszót a break kulcsszóval helyettesítjük.
- 8) Ettől eltekintve, egyetlen válasz sem helyes.

```
package gotopkg;

public class Main {

    public static void main(String[] args) {
        int határ = 0;
        for (;;) {
            System.out.println("a");
            if (határ++ > 10) {
                goto gotopkg.vege;
            }
        }
        gotopkg.vege:
        System.out.println("b");
    }
}
```

Laborkártyák

Melyik a helyes, miért?

- 1) Nem fordul le, mert A és B osztályok nem nyilvánosak.
- 2) Nem fordul le, mert a B osztály nem nyilvános, így nyilvánossá nem kiterieszthető.
- 3) Futása „Exception in thread "main" java.l
- 4) „A”-t.
- 5) „B”-t.
- 6) „C”-t.
- 7) „A”-t és „B”-t.
- 8) „A”-t és „B”-t és végül „C”-t.
- 9) Nem fordul le más okból.
- 10) Lefordul, de egy „A” kiírása után ClassCas

```
class A {  
    public void m() {  
        System.out.println("A");  
    }  
}  
  
class B extends A {  
    public void m() {  
        System.out.println("B");  
    }  
}  
  
public class Main extends B {  
  
    public void m() {  
        System.out.println("C");  
    }  
  
    public static void main(String[] args) {  
        .....  
        A main = (B) new Main();  
        main.m();  
    }  
}
```

Otthoni opcionális feladat

Saját Atan alapú RCSS csapat fejlesztése.



Kötelező olvasmány

Interfészek

NYJ

NYJ I/74-85

Csomagok

NYJ

NYJ I/97-105