



M Ű E G Y E T E M 1 7 8 2
Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Kollaboratív modellezés

Balogh Tímea III. évf., (BSc) mérnökinformatikus szakos hallgató

Konzulens: Debreceni Csaba doktorandusz, MIT

Rendszertervezés szakirány

Témalaboratórium összefoglaló

2016/17. I. félév

Modellvezérelt fejlesztés

Bizonyos informatikai rendszerek esetén kiemelt fontosságú a helyes működés, mert a hibák komoly anyagi károkat, vagy akár emberi tragédiákat is okozhatnak. (Pl. autó-, vasút- vagy repülőgép irányítási rendszerek) Az ilyen tulajdonságú rendszereket nevezzük biztonságkritikus rendszereknek. Ezek fejlesztésének modell alapú megközelítése ma már nagyon gyakori módszer. Ennek lényege, hogy a tervezés során magas absztrakciós szintű modelleket készítünk, majd ezeket finomítjuk tovább. Ez a komplexebb rendszerek átláthatóságát is nagyban megkönnyíti, a metodika legnagyobb előnye pedig az, hogy a modellekből automatikusan futtatható forráskódot generálhatunk, ezzel is minimalizálva az implementálás során előforduló emberi hibák számát.

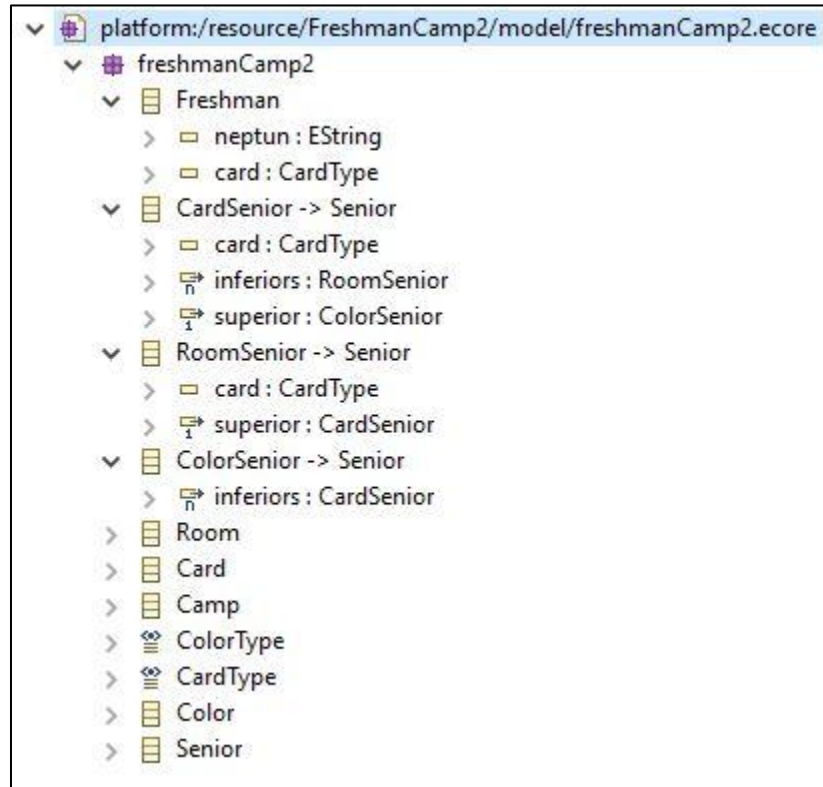
Eclipse Modeling Framework (EMF)

Az Eclipse szoftverkeretrendszer sajátossága, hogy funkciói pluginokba szerveződnek. Ezek telepítésével a fejlesztőkörnyezet kiterjeszthető úgy, hogy a Javán kívül más programnyelveket is támogasson, de léteznek pl. LaTeX és adatbáziskezelő pluginok is. Letölteni és telepíteni a Help/Install New Software... menüpontban lehet őket.

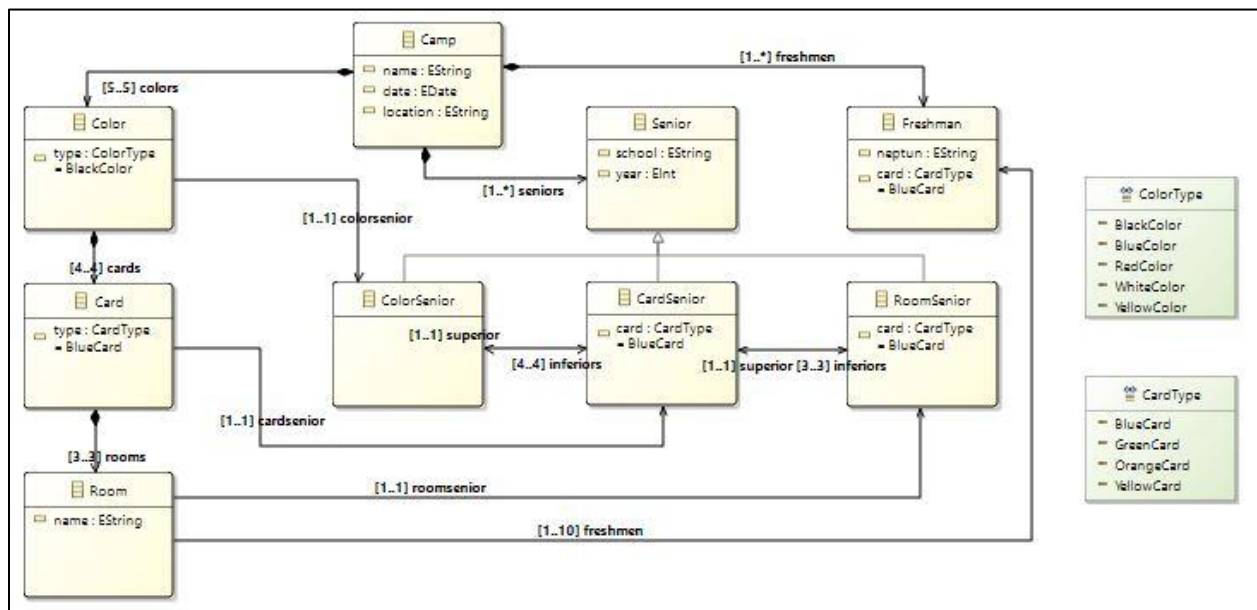
Az Eclipse Modeling Framework egy Eclipse pluginokból álló modellező és kódgeneráló keretrendszer. Megkülönbözteti a metamodellt (Ecore modell) a tényleges modelltől, előbbi a modell struktúráját írja le, utóbbi pedig a metamodell konkrét példánya.

A témalabor során az első feladatom egy EMF modell elkészítése volt, egy tutorial [1] segítségével az alább látható metamodellt hoztam létre. Az Ecore modellen található:

- EClass (pl. Camp, Freshman, Senior)
- EAttribute (pl. name, type, year)
- EEnum (ColorType, CardType)
- EReference (pl. Colorból colorseniorra (referencia), ColorSenior a Senior-ból (leszármazás), Color a Camp-ben (tartalmazás))

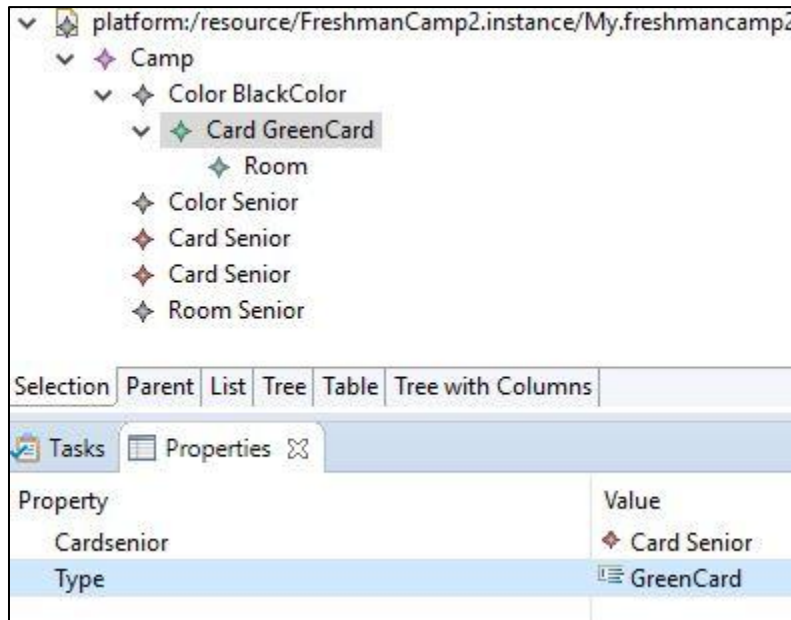


Metamodell (.ecore fájl)



Metamodell (.aird fájl)

A genmodel fájl alapján Java kódot, majd pedig plugineket generáltam a modellhez (*.edit, *.editor). Utóbbiak segítségével hoztam létre új modellpéldányt, ehhez a *.editor projectet futtattam a Run as/Eclipse Application paranccsal, ami egy új Eclipse példányt indít. Ebben egy varázsló segítségével készítettem el a példányt:



Példánymodell

Absztrakt és konkrét szintaxis

A modellek leírásához modellezési nyelveket használunk. Ezeknek részei az absztrakt szintaxis és a konkrét szintaxisok. Előbbi azt határozza meg, hogy a nyelvnek milyen típusú elemei vannak és ezek milyen kapcsolatban állnak egymással, vagyis ez maga a metamodel. Ehhez több konkrét szintaxis is megadható, ezek olyan szöveges vagy grafikus megjelenítést biztosítanak a modellhez, amiktől olvashatóvá és szerkeszthetővé válik a modell leírása. A szöveges nyelvek használatának előnye, hogy a modell könnyebben építhető vele, modellek olvasása viszont a grafikus nyelvek segítségével egyszerűbb.

A következő feladatom egy szöveges konkrét szintaxis készítése volt az előbbi EMF modellemhez az Xtext keretrendszer segítségével. Az Xtext Eclipse-be való telepítése után több tutorial [2][3][4] alapján hoztam létre egy Xtext Project-et és definiáltam egy nyelvtant a modellemhez. (Megjegyzés: meglévő metamodelból is generálható nyelvtan, ehhez a következő típusú projectet kell létrehozni: Xtext Project From Existing Ecore Models.)

```
Camp :
'camp' name=ID
'colorseniors' '{'
  colorseniors+=ColorSenior*
'}'
'cardseniors' '{'
  cardseniors+=CardSenior*
'}'
'roomseniors' '{'
  roomseniors+=RoomSenior*
'}'
'colors' '{'
  colors+=Color*
'}'
;

Freshman :
name=Name '{'
  'neptun' ':' neptun=STRING
  'color' ':' color=ColorType
  'card' ':' card=CardType
'}'
;

ColorSenior :
name=Name '{'
  'year' ':' year=INT
  'color' ':' color=ColorType
  'inferiors' ':' inferiors+=[CardSenior|Name] (',' inferiors+=[CardSenior|Name]) *
'}'
;

CardSenior :
name=Name '{'
  'year' ':' year=INT
  'color' ':' color=ColorType
  'card' ':' card=CardType
  'superior' ':' superior+=[ColorSenior|Name]
  'inferiors' ':' inferiors+=[RoomSenior|Name] (',' inferiors+=[RoomSenior|Name]) *
'}'
;

RoomSenior :
name=Name '{'
  'year' ':' year=INT
  'color' ':' color=ColorType
  'card' ':' card=CardType
  'superior' ':' superior+=[CardSenior|Name]
'}'
;
```

Xtext nyelvtan (részletek)

A nyelvtanon belül a Generate Xtext Artifacts parancsra kattintva létrejönnek a megfelelő pluginek. A nyelvtant tartalmazó projectből egy új Eclipse példányt indítva létrehoztam egy olyan kiterjesztésű fájlt, amit a project létrehozásakor megadtam. Ebben szerkeszthető a modell.

```
cardseniors {
  Timi {
    year : 2014
    color : BlueColor
    card : OrangeCard
    superior : Nur
    inferiors : Aron, Bazsi, Abus
  }
}

roomseniors {
  Aron {
    year : 2015
    color : BlueColor
    card : OrangeCard
    superior : Timi
  }
}
```

- ▼ ♦ Camp Tata2016
 - ♦ Color Senior Nur
 - ♦ Card Senior Timi
 - ♦ Room Senior Aron
 - ♦ Room Senior Bazsi
 - ♦ Room Senior Abus
 - ▼ ♦ Color Blue
 - ▼ ♦ Card orange
 - ♦ Room firstBlue
 - ♦ Room secondBlue
 - ♦ Room thirdBlue

Szöveges és tree editorok

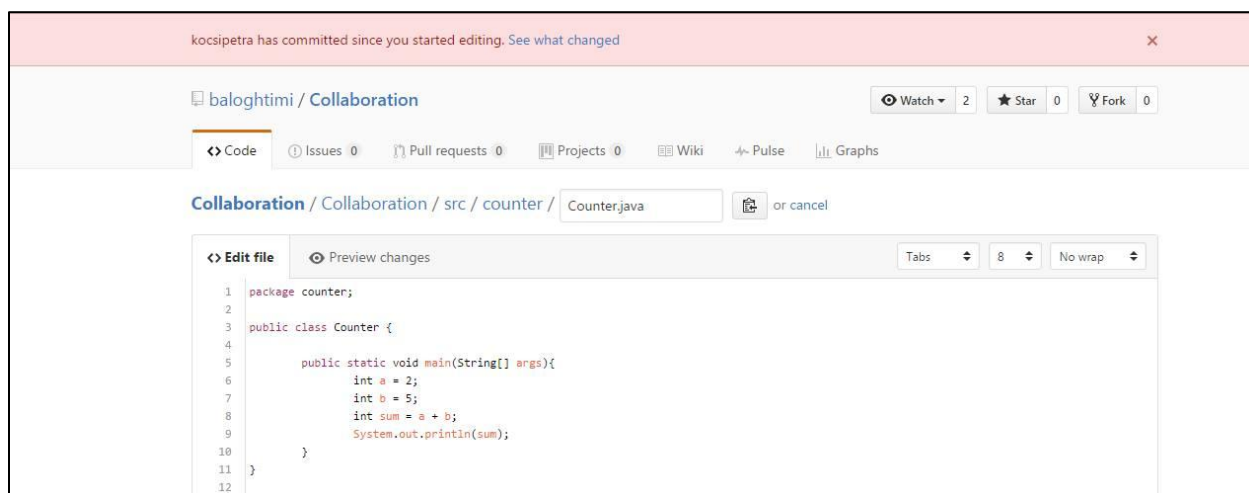
Kollaboráció

A modellvezérelt fejlesztés során is gyakori, hogy egy projecten egyszerre többen dolgoznak akár időben és térben is elosztottan. A közös munkán végrehajtott különböző változtatásokat össze kell egyeztetni úgy, hogy az eredmény mindenki számára konzisztens legyen. Ebben különféle kollaborációs eszközök segítik a csapatban dolgozókat.

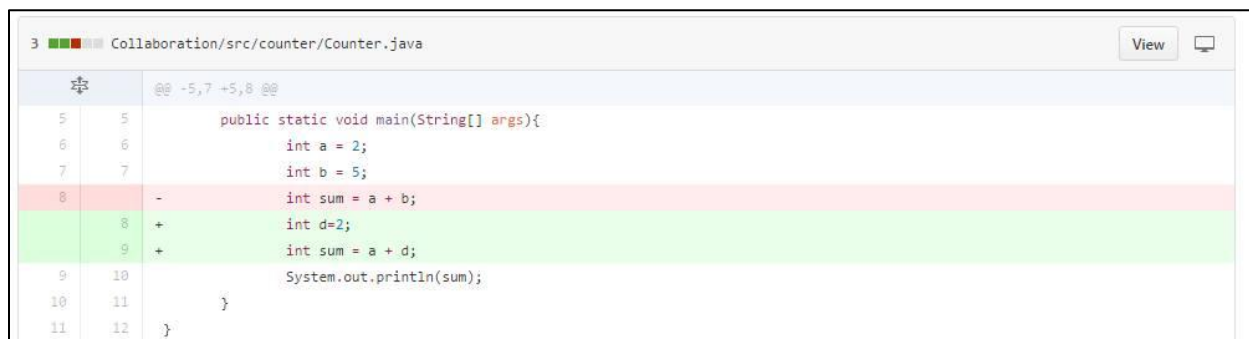
Ezeket a kollaborációs eszközöket csoportosíthatjuk online vagy offline jellegük szerint. Online kollaboráció esetén a dokumentumot vagy a modellt a résztvevők egy időben módosítják, a változások azonnal érvényre jutnak, vagyis konfliktusok nem keletkeznek. Offline esetben mindenki egy saját példánnyal rendelkezik, ezt módosítja, majd visszatölti a közös szerverre, ahol összefésülődnek a különböző verziók. Ebben az esetben viszont a változatok között adódnak konfliktusok, amelyeket valamilyen szabályok alapján fel kell oldani.

Git/Github – offline, szöveg alapú

A szövegalapú kollaborációs technológiák közül a Gitet/Githubot próbáltuk ki. Ide töltöttük fel az elkészült projectjeinket, valamint kipróbáltuk, hogy ezek az eszközök hogyan kezelik a konfliktusok feloldását. Ha két felhasználó egyszerre kezdi módosítani ugyanazt a fájlt, és az egyikőjük commitolja, akkor a másikat az ő commitjakor értesíti a rendszer, hogy a szerkesztés megkezdése óta a fájl módosult. A konfliktus feloldása a felhasználó dolga, ezután a rendszer az ő commitját is érvényre juttatja.



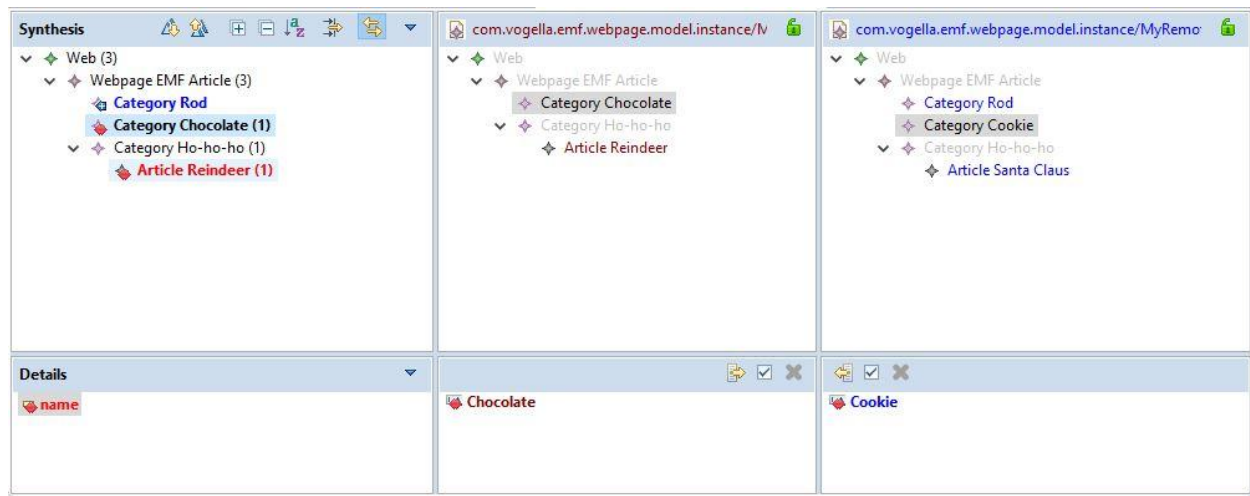
Github figyelmeztetése, hogy a fájl megváltozott



A konfliktus megjelenítése Githubon

EMF Diff/Merge – offline, modell alapú

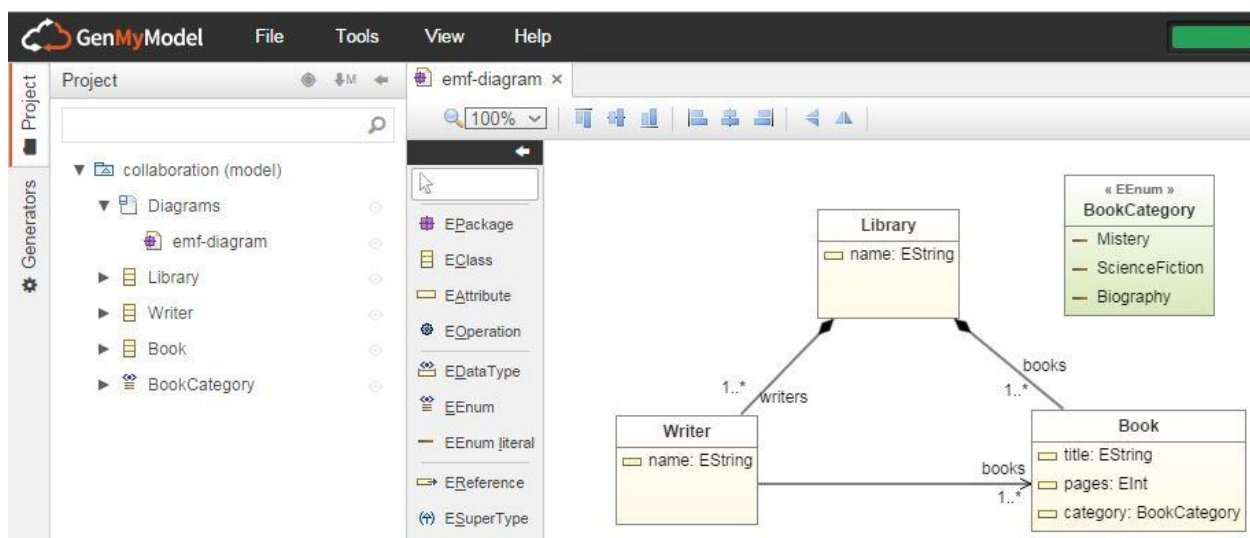
Ugyanúgy ahogy szövegeket, modelleket is összehasonlíthatunk és összefésülhetünk, csak más szabályok alapján. Az offline modell alapú kollaborációs eszközök közül az EMF Diff/Merge-t próbáltam ki. Ehhez először engedélyeznem kellett EMF-ben a UUID-kat, ehhez egy tutorial[5] lépéseit követtem. Ezután létrehoztam egy egyszerű modell példányát, majd ennek még két másolatát, amelyeket úgy módosítottam, hogy ütközés keletkezzen közöttük. Az eszközt alkalmazva megjelentek ezek a konfliktusok: baloldalon látható az eredeti modell és a másik kettőtől való eltérések. (Pirossal a bal oldali, kézzel a jobb oldali modellek elemei.) Az alsó sávban a kijelölt attribútumok eltérő értékei láthatók. Ki lehet választani, hogy a megjelölt változások közül melyek jussanak érvényre, majd a program összefésüli a modelleket.



EMF Diff/Merge példányokat összehasonlító ablaka

GenMyModel – online, modell alapú

Az offline modell alapú eszközöktől abban különbözik, hogy ez egy web alapú online technológia, ami azt jelenti, hogy a felhasználók egyszerre egy időben szerkeszthetik a modellt és a változások azonnal megjelennek mindenkinél.



GenMyModel felhasználói felülete

Források:

- [1] <http://www.vogella.com/tutorials/EclipseEMF/article.html>
- [2] http://www.eclipse.org/Xtext/documentation/101_five_minutes.html
- [3] <http://blog.efftinge.de/2009/11/xtext-using-existing-ecore-models.html>
- [4] <http://www.eclipse.org/articles/article.php?file=Article-BuildYourOwnDSL/index.html>
- [5] <http://eioki.eu/2007/03/05/how-to-enable-uuid-in-emf-generated-model-to-get-copypaste-working>