



IEGULDĪJUMS TAVĀ NĀKOTNĒ!

Eiropas Reģionālās attīstības fonds

Prioritāte: 2.1. Zinātne un inovācijas

Pasākums: 2.1.1. Zinātne, pētniecība un attīstība

Aktivitāte: 2.1.1.1. Atbalsts zinātnei un pētniecībai

Projekts: "Multi - modeļu izstrādes tehnoloģija .NET pielietojumu projektiem"

Projekta sākuma datums: 2014.gada 1.janvāris.

Projekta beigu datums: 2015.gada 30.jūnijs.

Līguma Nr. 2013/0031/2DP/2.1.1.1.0/13/APIA/VIAA/010

ESF finansējuma saņēmējs: SIA, SWH SETS

Sadarbības partneris: Elektronikas un datorzinātņu institūts (EDI)

Projekta aktivitātes Nr.2.2 "Universālā modeļa redaktora izstrāde" progresa pārskats

Pārskats Nr.16. par periodu no 2014.gada 1.jūlija līdz 2014.gada 31.decembrim.

SATURS

1.	Kopsavilkums	3
2.	Ievads	4
3.	Universālā modeļa redaktora izstrādes pamatprincipi	5
3.1.	Vajadzības analīze	5
3.2.	Prasības	5
3.3.	Izstrādes vide	5
4.	Redaktora struktūra	6
4.1.	Saraksta daļa	7
4.2.	Tabulas daļa	7
4.3.	Skatu veidi	8
5.	Darbs ar redaktoru	9
5.1.	Palaišana	9
5.2.	Uzstādījumi	9
5.2.1.	Metamodeļa ceļš	9
5.2.2.	Skatu veidi	10
5.3.	Modeļa rediģēšana	10
5.3.1.	Jauna modeļa izveidošana	10
5.3.2.	Modeļa ielādēšana un rediģēšana	11
5.3.3.	Modeļa saglabāšana	11
5.3.4.	Modeļa lokalizācija	11
5.4.	Papildus iespējas	13
5.4.1.	Eksports uz CSV failu	13
5.4.2.	Modeļa verificēšana	13
5.4.3.	Xmmedus meta modeļa faila izveidošana	13
6.	Rezultāti	14
7.	Literatūras saraksts	15

1. Kopsavilkums

Pārskata periodā (2014-07-01 – 2014-12-31.) projekta „Multi - modeļu izstrādes tehnoloģija .NET pielietojumu projektiem” aktivitātes Nr.2.2 "Universālā modeļa redaktora izstrāde" ietvaros veikti šādi darbi:

1. Pilnveidota redaktora struktūra – saraksta daļa, tabulas daļa un skatu veidi;
2. Modeļa redaktora uzstādījumu funkciju izstrāde;
3. Modeļa rediģēšanas funkcijas – jauna modeļa izveidošana, modeļa ielādēšana un rediģēšana, modeļu saglabāšana, modeļa lokalizācija – izstrāde;
4. Papildus funkcijas – Eksports uz CSV failu, modeļa verificēšana Xmmmedus meta modeļa faila izveidošana – izstrāde.
5. Aktivitātes pētnieciskā darbība apspriesta ik nedēļas projekta semināros.

2. Ievads

Šis pārskats ir veltīts projekta apakšaktivitātes Nr.2.2 "Universālā modeļa redaktora izstrāde" ietvaros paveiktajam. Apakšaktivitātes ietvaros tika izstrādāts Universālais modeļa redaktors.

Pārskata pirmajā daļā ir aprakstīti universālā modeļu redaktora izstrādes pamatprincipi un izstrādes vide. Otrā daļa ir apraksta modeļu redaktora struktūra. Trešajā – universālā modeļu redaktora iespējas un funkcionalitāte.

3. Universālā modeļa redaktora izstrādes pamatprincipi

3.1. Vajadzības analīze

Projekta apakšaktivitātes Nr.1.4 "Modeļu kodēšana un ielādes metožu izpēte" ietvaros, tika izstrādāts konkrēts modeļu apmaiņas formāta kodēšanas veids. Šis konkrētais formāts dod iespēju aprakstīt un glabāt modeļa informāciju XML formātā. Apraksts un piemērs šim formātam tiek dots projekta aktivitātes Nr.1.4 "Modeļu kodēšana un ielādes metožu izpēte" progresā pārskatā [1] 5. nodaļā.

Galvenā priekšrocība šim formātam ir viņa universalitāte, kas ļauj aprakstīt modeļus, bāzētus uz dažādiem metamodeļiem. Lai korekti aprakstītu modeli jāpārzina tam specifiska metamodeļa jēdzieni, to atribūti un savstarpējas saites, kā arī XML pierakstu. Tam varētu izmantot teksta redaktoru, bet darbietilpība un kļūdu varbūtība šim variantam ir pārāk liela. Šī procesa sarežģītības pakāpe aug vēl vairāk, ja ir vairāki metamodeļi, un/vai modelis ir liels. Acīmredzami, pacēlās jautājums par redaktoru, kas atbalstītu mūsu XML notāciju un atļautu aprakstīt modeļus, bāzētus uz dažādiem metamodeļiem, lietotājam draudzīgajā vidē.

3.2. Prasības

Pēc vajadzības analīzes tika izvirzītās sekojošas prasības modeļu redaktoram:

Tehniskas:

- jābūt iespējai taisīt modeli, balstīto uz dažādiem, lietotāja definētiem meta modeļiem;
- jābūt iespējai ielādēt meta modeli gan .dll formātā, gan .mmedus formātā;
- redaktoram jābūt balstītam uz meta meta modeļa jēdzieniem, kas nodrošinātu universalitāti;
- rādīt/manipulēt ar modeļa datiem izvēlētajā meta modeļa terminos;
- jābūt iespējai saglabāt ievadīto modeli mūsu izveidotajā XML formātā.

Lietotāja un GUI:

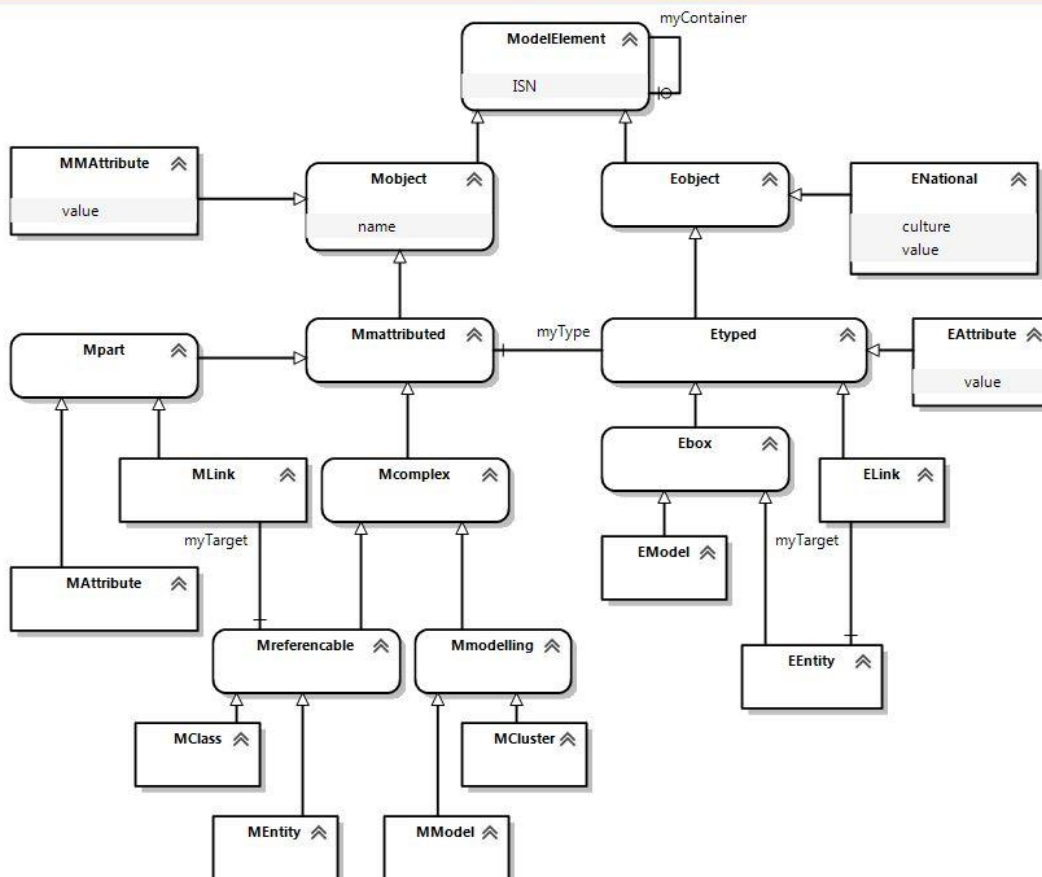
- jānodrošina lietotājam draudzīgo grafisko vidi modeļu manipulēšanai (vēsture rādīja, ka tabulveida ievads ir visefektīvākais modeļa rediģēšanā);
- jānodrošina iespēja ielādēt, parādīt, izmainīt un saglabāt modeli.

3.3. Izstrādes vide

Redaktora izstrādei ir izvēlēts Microsoft .NET Framework un WPF tehnoloģija GUI realizācijai.

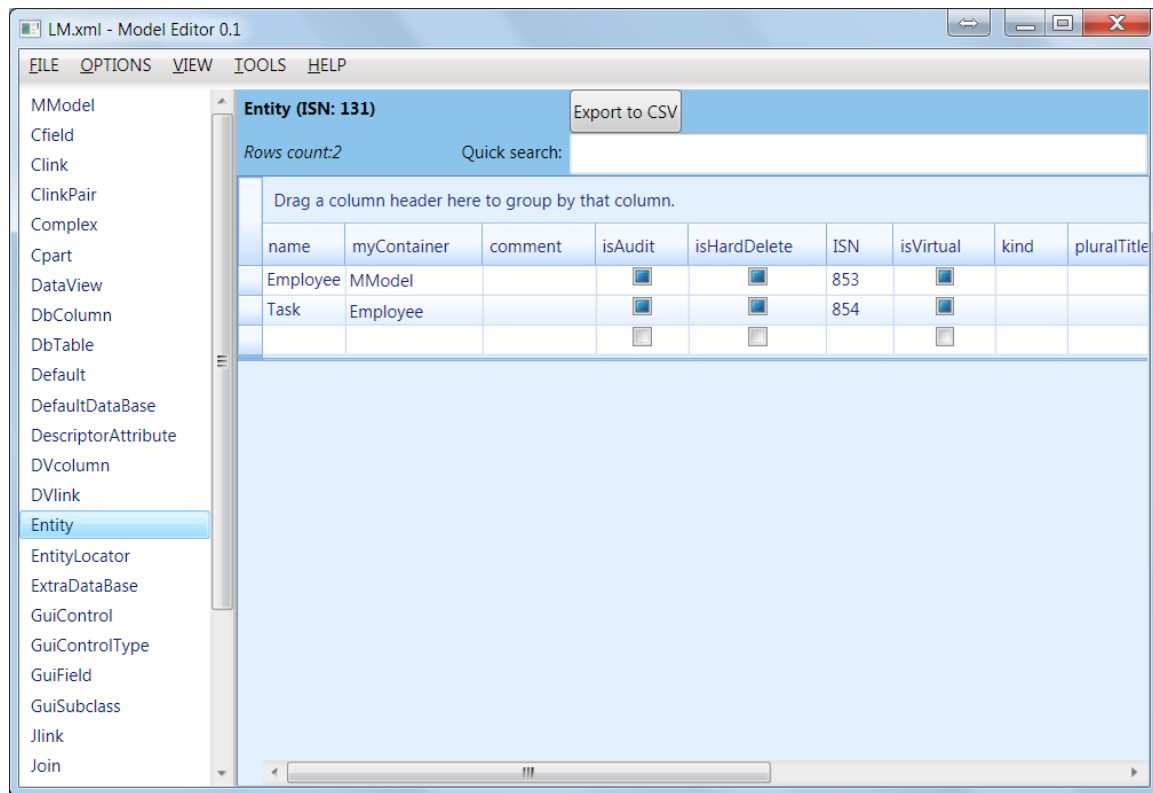
4. Redaktora struktūra

Redaktora pamatā ir meta meta modeļa jēdzieni, kas nemainās, pārejot no viena meta modeļa un citu. Tas dod iespēju neierobežot redaktoru ar meta modeļa saturu. Modeļa objekta papildināšanai ir izmantots API, kurš aprakstīts projekta apakšaktivitātes Nr.2.3.1 "Elementu pievienošanas izstrāde" pārskatā [2].



Zīm. 1.

Modeļa glabāšanas modelis (Zīm. 1), kas ir aprakstīts pārskatā [2], dod priekšstatu par meta metamodeļa terminiem, uz kuru pamata ir uzbūvēts redaktors. Bet darbošanās universālajā meta meta modeļa līmenī ir ļoti tehnoloģiski sarežģīta. Līdz ar to, lai dotu izstrādātājam iespēju darboties meta modeļa jēdzienos redaktors strādā ar D-modeļu interfeisiem [3]. Universālais redaktors sastāv no divām pamatpaneļiem (Zīm. 2) - Saraksta daļa (kreisajā pusē) un Tabulas daļa (labajā pusē).



Zīm. 2.

4.1. Saraksta daļa

Saraksta daļa ir saraksts no visiem izvēlētā metamodeļa MEntity objektiem un Mmodel objekts, kas kalpos par modeļa saknes objektu (sk. [2]). Citiem vārdiem, saraksts atspoguļo objektu tipus, kurus atbalstā izvēlētais metamodelis. Ir pieņemts, ka pārejot starp šim objektu tipiem, labajā pusē parādās tabula ar izvēlēta tipa objektiem (Zīm. 2).

4.2. Tabulas daļa

Tabulas daļa ir kreisajā pusē izvēlētā tipa objektu saraksts ar viņu atribūtiem - modeļa apraksts (Zīm. 2). Katra tabulas kolona pārstāv izvēlētā MEntity objekta MAttribute vai MLink objektu, kur kolonas nosaukums ir attiecīgā MAttribute vai MLink nosaukums. Tabulas rindas pārstāv izvēlētā MEntity tipa EEntity objektus, kur katrā cellē ir atbilstoša atribūta vērtība.

Ir pieņemts, ka neatkarīgi no meta modeļa, katram modeļa objektam ir jāsaturs specifiskie iebūvētie atribūti: ISN, myContainer, id un refName.

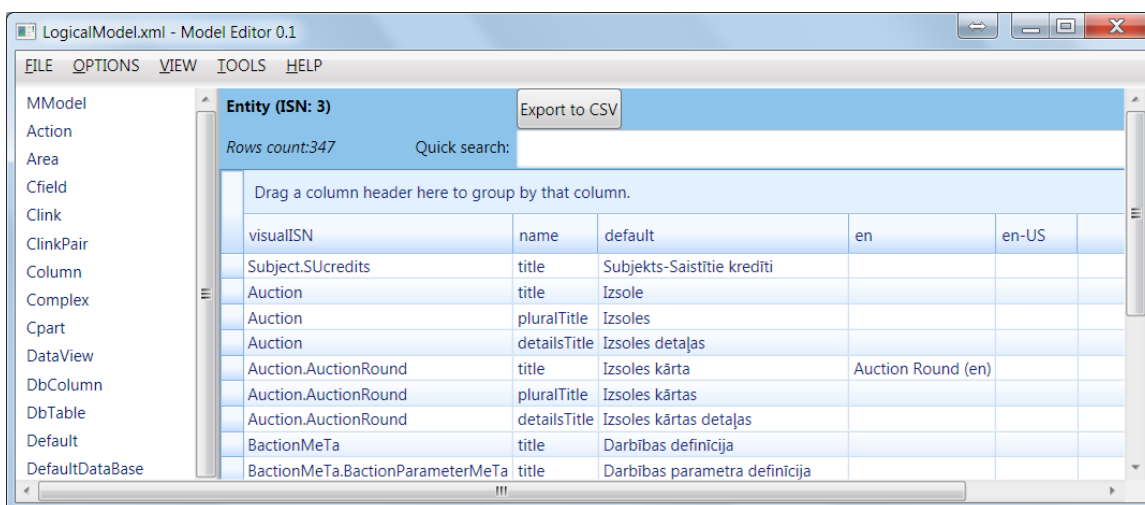
Tabulas daļas augšā ir norādīti tekošā objekta tips ar iekšējo identifikatoru (ISN) un tā tipa objektu skaits.

Tabulas daļa arī nodrošina tabulas kolonas paslēpšanu/parādīšanu (Hide/Unhide), kārtošānu pēc jebkuras kolonas un modeļa datu filtrēšanu pēc kolonu satura.

4.3. Skatu veidi

Metamodelis bieži vien satur modeļa objektu atribūtus, kas kalpo tīri tehniskiem (iekšējai realizācijai specifiskiem) nolūkiem. Atribūts `isTechnical` šim atribūtiem meta modelī ir `True`. Šajā gadījumā, lai atvieglotu gala lietotājam darbu, un dotu iespēju koncentrēties uz biznesa vajadzībām, redaktors atbalsta dažādus galvenā loga skatus:

1. “Entity” skats – katram objektu tipam kreisajā pusē tabulā rāda tikai atribūtus, kuri nav atzīmēti kā tehniskie (Zīm. 2)(`isTechnical=false`);
2. “Technical” skats - katram objektu tipam kreisajā pusē tabulā rāda visus atribūtus, tajā skaitā arī tehniskie;
3. “Culture” skats - specifisks skats, lai atvieglotu tekstu tipa atribūtu vērtību tulkošanu dažādās valodās (Zīm.3).



Zīm.3.

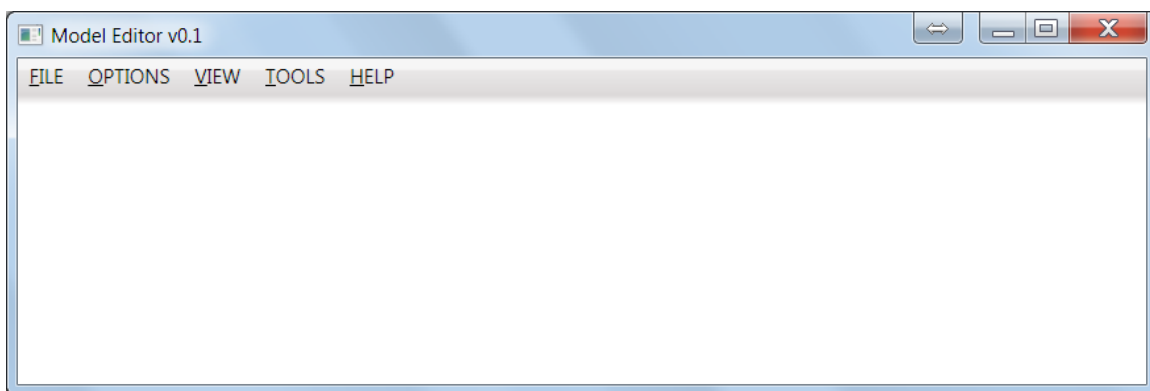
5. Darbs ar redaktoru

5.1. Palaišana

Priekšnosacījumi redaktora lietošanai:

- OS – Windows 7, vai jaunāka;
- Tiesības .exe failu palaišanai.

Redaktora lietošanai nav vajadzīga instalēšana, ir vienkārši jāpalaiž MEditor.exe fails. Zīmējumā 4 ir redaktora galvenais logs ar izvēlni. Pie palaišanas, tajā pašā direktoriņā, kurā atrodas Meditor.exe fails tiek izveidots MEditor.settings fails ar redaktora uzstādījumiem.

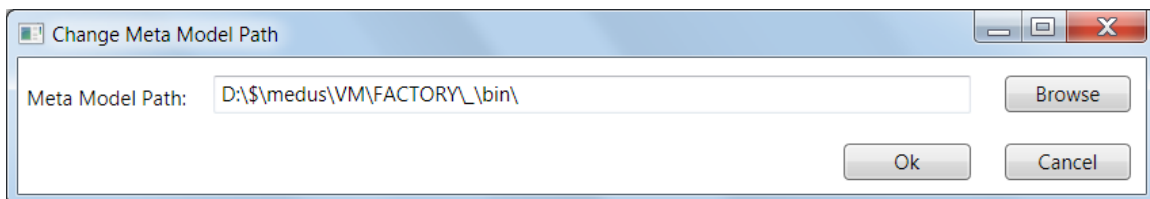


Zīm.4.

5.2. Uzstādījumi

5.2.1. Metamodeļa ceļš

Pirms rediģēt esošo, jābūt pareizi uzstādītam meta modeļa ceļam (ceļš, kur atrodas meta modeļa fails). Katrs modeļa fails satur sevī sava meta modeļa faila nosaukumu, kurš tiks meklēts zem norādītā metamodeļa ceļa. Lai to izdarītu, jāizvēlas izvēlnes punktu “OPTIONS->Change MetaModel Path” (Zīm.5).



Zīm. 5.

Poga “Browse” atver “Browse for Folder” logu, kur var izvēlēties vajadzīgo ceļu un nospiegt “OK”. Tad ceļš parādās laukā “Meta Model Path”. Šajā laukā ceļu var ievadīt arī ar roku vai nokopējot. Lai saglabātu ceļu, jānospiež poga “OK”. Poga “Cancel” atceļ visas izmaiņas. Pēc noklusējuma, Metamodeļa ceļš rāda uz direktoriju, kur atrodas MEditor.exe

fails. Metamodeļa ceļš tiek saglabāts uzstādījumu failā, lai viņu nav atkārtoti jādefinē pie katras redaktora palaišanas.

5.2.2. Skatu veidi

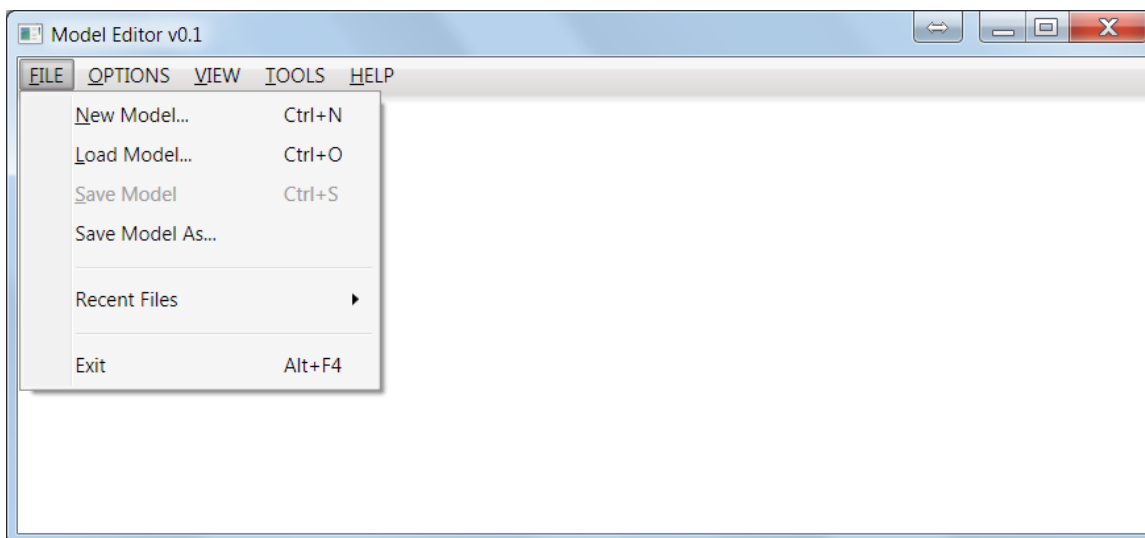
Izvēlnes punkts “View” atļauj pārslēgt saktu veidus (Entity, Technical, Culture), aprakstītus šī pārskata 4.3. daļa.

5.3. Modeļa rediģēšana

Izvēlnes punkts “File” nodrošina modeļa rediģēšanas funkcionalitātes izsaukumu (Zīm.6)

“Recent File” apakšizvēlne rāda sarakstu ar visbiežāk rediģējamiem modeļa failiem un ļauj ielādēt vienu no šī saraksta bez faila papildmeklēšanas.

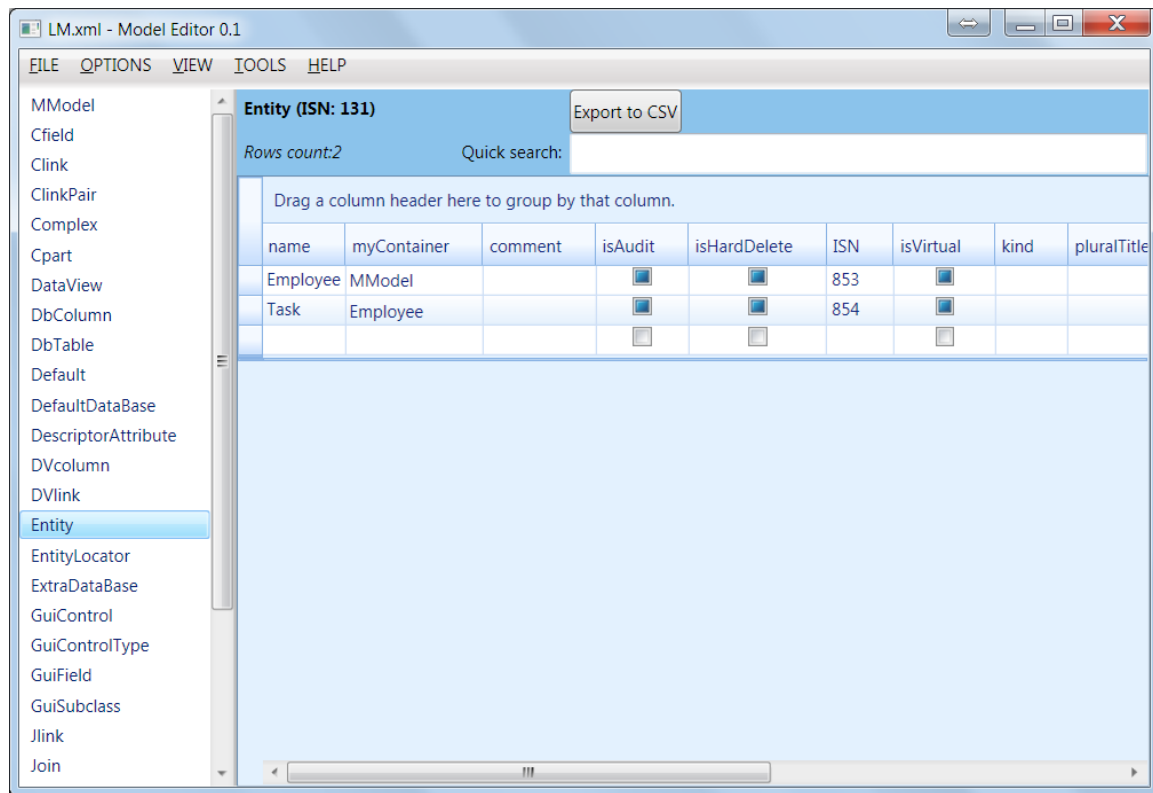
“Exit” izvēlnes punkts aizver redaktoru, pārprasot, vai gribas saglabāt izmaiņas, ja ir tādas nesaglabātas.



Zīm. 6.

5.3.1. Jauna modeļa izveidošana

Lai uztaisītu jaunu modeli, jāizvēlas izvēlnes punkts “New Model”, ar kura palīdzību, izvelieties bāzes meta modeli - .dll vai .mmedus, .cmedus vai .umedus failu un nospiediet “OK”. Tiks atvērta jauna modeļa logs. Pārejot starp objektu tipiem labajā pusē, kreisajā pusē, tabulā, jāievada celles vērtības, tāda veida veidojot jaunus modeļa objektus. Lai tikko ievadīta celles vērtība saglabātos, jāuzklikšķina uz jebkuras citas celles (Zīm. 7). Pēc rediģēšanas, modelis jāsavienā, izmantojot izvēlnes punktu “Save” (sk. 5.3.3)



Zīm. 7.

5.3.2. Modeļa ielādēšana un rediģēšana

Piezīmes: Pirms modeļa faila ielādēšanas, pārliecinieties, ka ir pareizi uzstādīts MetaModeļa ceļš (sk. 5.2.1).

“Load Model” izvēlnes punkts dod iespēju izvēlēties un atvērt modeļa failu. Tālāk faila saturu var rediģēt līdzīgi jaunajam failam (sk. 5.3.1)

5.3.3. Modeļa saglabāšana

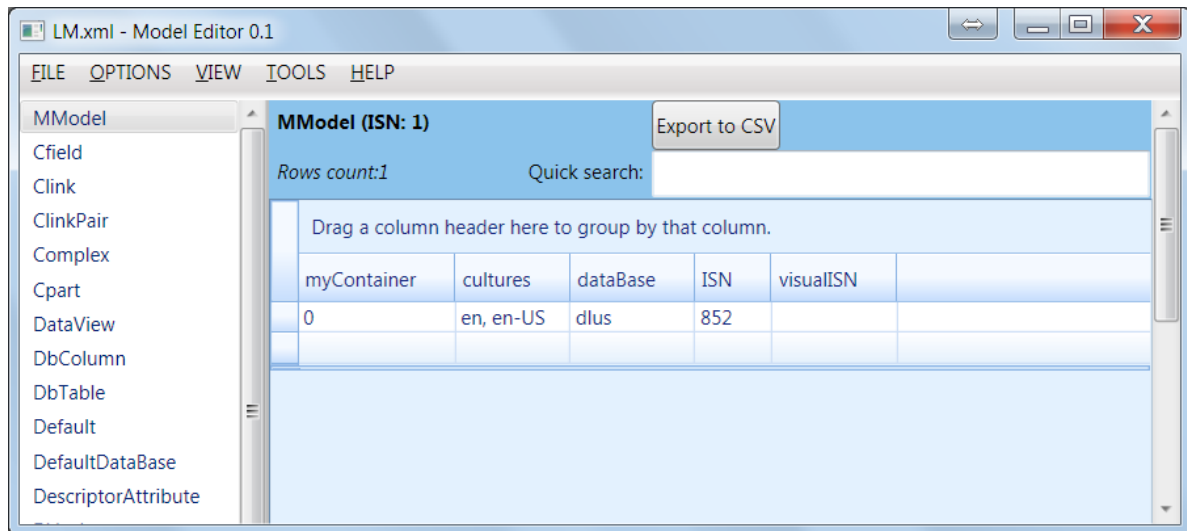
Lai saglabātu modeli failā, jāizvēlas “Save” vai “Save As” izvēlnes punkts, kas atļauj attiecīgi saglabāt modeli esošajā failā vai citā, lietotāja norādītāja failā.

5.3.4. Modeļa lokalizācija

“Culture” redaktora skatu veids atbalsta modeļa rediģēšanu, saistīto ar modeļa datu lokalizāciju un tulkošanu. “Entity” redaktora skats dod iespēju ievadīt teksta vērtības tikai vienā valodā. Šī valoda ir uzskatāma par noklusējama valodu modeļa datu vērtībām.

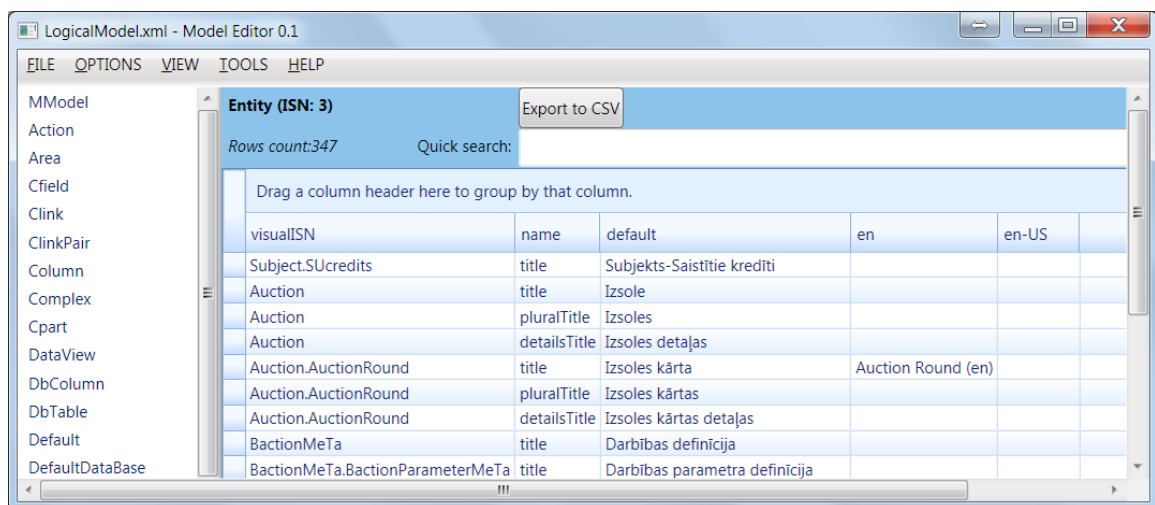
Bet ļoti bieži ir nepieciešamība atbalstīt vairākas kultūras (valodas) vienā modeļa ietvaros. Mūsu redaktors dod tādu iespēju – “Entity” skatā kreisajā pusē jāizvēlas objektu tips MModel un labajā pusē jānodēfinē atribūts “cultures”(Zīm.8) – saraksts ar kultūrām,

piemēram: “en, en-US”. Kultūras ir jānorāda Windows Operating System un Microsoft .NET Framework atbalstāmajā formātā [4].



Zīm.8.

Tagad, lai ievadītu tulkojumus, jāpārslēdz redaktora skats uz “Culture”(Zim. 9).



Zīm.9.

Te kreisajā pusē ir tāds pats objektu tipu saraksts kā “Entity” skatā, bet labajā pusē tabula satur sekojošo:

- pirmā kolona nav rediģējama un satur objekta nosaukumu;
- otrā kolona nav rediģējama un satur objekta teksta atribūta nosaukumu;
- trešā kolona nav rediģējama un satur objekta atribūta vērtību kultūrā (valodā) pēc noklusējuma.

- tālāk nāk kolonas, atbilstoši augšminētam MModel objekta atribūta “cultures” vērtībai. Katrā no šīm kolonām var ievadīt atbilstošo tulkojumu un saglabāt modeli.

5.4. Papildus iespējas

Izvēlnes punkts “Tools” nodrošina papildus funkcionalitātes izsaukumu. Lai izsauktu šo funkcionalitāti ir izmantots modeļa ietvarā definētais API.

5.4.1. Eksports uz CSV failu

Lai eksportētu izvēlēto tabulu uz CSV failu, jānospiež poga “Export To CSV” Tabulas skatā augšā, jānodefinē faila nosaukums un jānospiež “OK”. Tiks izveidots fails ar norādīto nosaukumu.

5.4.2. Modeļa verificēšana

“Verify Model” izvēlnes punkts nodrošina modeļa pārbaudi, atbilstoši konkrētā meta modelī ieprogrammētiem noteikumiem (ir pieejams tikai, ja meta modelis ir .dll formātā) un kļūdu gadījumā izdod sarakstu ar kļūdām un to vietu.. Piemēram, vai visas entītijas nosaukumi vienas datubāzes ietvaros ir unikāli.

5.4.3. Xmmetus meta modeļa faila izveidošana

“Create Xmmetus metamodel” izvēlnes punkts dod iespēju izveidot kopīgo meta modeli, no meta modeļiem vienā hierarhijā.

6. Rezultāti

Aktivitātes ietvaros tika izstrādāts universālais modeļa redaktors, kas atvieglo modeļa izveidošanu. Redaktors atbalsta jebkuru, bāzētu uz meta meta modeļa, lietotāja definēto meta modeli, ka bāzi priekš modeļa. Pašreizējā redakcijā redaktors realizē tabulveida lietotāja saskarni. Redaktora lietošana parādīja, ka redaktors varētu atbalstīt arī kokveida lietotāja saskarne, kas ir ērti dažos gadījumos, piemēram, konkrēta modeļa objekta meklēšanā.

7. Literatūras saraksts

1. Nr.1.4 "Modeļu kodēšana un ielādes metožu izpēte" progresā pārskatā
2. Nr.2.3.1 "Elementu pievienošanas izstrāde" pārskatā
3. Nr.2.3.2 "Modeļa importa izveide" progresā pārskats
4. [http://msdn.microsoft.com/en-us/library/system.globalization.cultureinfo\(v=vs.110\).aspx](http://msdn.microsoft.com/en-us/library/system.globalization.cultureinfo(v=vs.110).aspx)