



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.1.3.1 "Navigācijas nodrošināšana" progresa pārskats

Pārskats Nr. 3 par periodu no 2014.gada 1.janvāra līdz 2014.gada 30.jūnijam.

SATURS

1.	Kopsavilkums.....	4
2.	Ievads.....	5
3.	Navigācijas modelis.....	6
4.	Navigācijas funkcijas.....	8
4.1.	Model (datu glabātuve).....	8
4.1.1.	ModelElement get(int key).....	8
4.1.2.	ModelElement this[int key].....	8
4.1.3.	IEnumerable<ModelElement> parts(int dadISN).....	8
4.1.4.	IEnumerable<T> tPartsQ<T>(int dadISN).....	8
4.1.5.	List<T> tParts<T>(int dadISN).....	8
4.1.6.	MModel mmodel.....	9
4.1.7.	List<MEntity> MEntityS.....	9
4.1.8.	List<MClass> MClassS.....	9
4.1.9.	List<Etyped> parts(int dadISN, int partTypeISN).....	9
4.1.10.	List<Etyped> parts(int dadISN, List<int> partTypeISNs).....	9
4.1.11.	IEnumerable<T> typed<T>(IEnumerable<ModelElement> from).....	9
4.1.12.	string hardcodedBase.....	9
4.2.	ModelElement.....	10
4.2.1.	int myContainer.....	10
4.2.2.	List<T> getParts<T>(Model store).....	10
4.3.	Mattributed.....	10
4.3.1.	IEnumerable<MMAAttribute> MMAAttributeS(Model store).....	10
4.3.2.	List<MMAAttribute> MMAAttributeM(Model store, string name).....	10
4.3.3.	MMAAttribute MMAAttributeF(Model store, string name).....	10
4.3.4.	string MMAAttributeV(Model store, string name).....	11
4.3.5.	bool MMAAttributeB(Model store, string name).....	11
4.3.6.	bool isTechnical(Model store).....	11
4.4.	Mpart.....	11
4.4.1.	bool isMultiple(Model store).....	11
4.4.2.	bool isMandatory(Model store).....	11
4.5.	MAttribute (metaatribūts).....	12
4.5.1.	string dataType(Model store).....	12
4.5.2.	public bool isName(Model store).....	12
4.5.3.	bool isNational(Model store).....	12
4.5.4.	string constraint(Model store).....	12
4.6.	MLink (metanorāde).....	12
4.6.1.	int myTarget.....	12
4.6.2.	Mreferencable myTargetO(Model store).....	12
4.7.	Mcomplex.....	13
4.7.1.	int mySuperClass.....	13
	Līguma Nr. 2013/0031/2DP/2.1.1.1.0/13/APIA/VIAA/010	2

4.7.2.	List<int> myExtraSuperClasses.....	13
4.7.3.	List<int> family(Model store, FamilyKind kind).....	13
4.7.4.	List<T> metaPartS<T>(Model store, FamilyKind).....	13
4.7.5.	List<MAttribute> MAttributeS(MEDUS.Model store).....	13
4.7.6.	List<MAttribute> MAttributeSs(Model store).....	14
4.7.7.	List<MLink> MLinkS(Model store).....	14
4.7.8.	List<MLink> MLinkS(Model store).....	14
4.7.9.	int myNameAttribute(Model store).....	14
4.7.10.	bool isComplexVisualISN(Model store).....	14
4.8.	Mreferencable.....	14
4.8.1.	List<int> eTypes(Model store).....	14
4.9.	Etyped.....	15
4.9.1.	int myType.....	15
4.9.2.	Mattributed myTypeO(Model store).....	15
4.9.3.	string myTypeN(Model store).....	15
4.10.	ELink (modeļa norāde).....	15
4.10.1.	int myTarget.....	15
5.	Literatūras saraksts.....	16

1. Kopsavilkums

Pārskata periodā (2014-01-01 – 2014-06-30) projekta „Multi - modeļu izstrādes tehnoloģija .NET pielietojumu projektiem” aktivitātes Nr.1.3.1 "Navigācijas nodrošināšana" ietvaros veikti šādi darbi:

1. Izstrādāta klase Model - datu glabātuves navigācija, kas ietver atbilstošo algoritmu un funkciju izstrādi.
2. Izstrādāta klase ModelElement, kas ietver funkciju myContainer un getPartT izstrādi.
3. Izstrādāta klase Mattributed un tās funkcijas.
4. Izstrādāta klase Mpart un tās funkcijas.
5. Izstrādāta klase MAttribute un tās funkcijas.
6. Izstrādāta klase MLink un tās funkcijas.
7. Izstrādāta klase Mcomplex un tās funkcijas.
8. Izstrādāta klase Mreferencable un tās funkcija eTypes.
9. Izstrādāta klase Etyped un tās funkcijas.
10. Izstrādāta klase Elink un tā funkcija myTarget.

2. Ievads

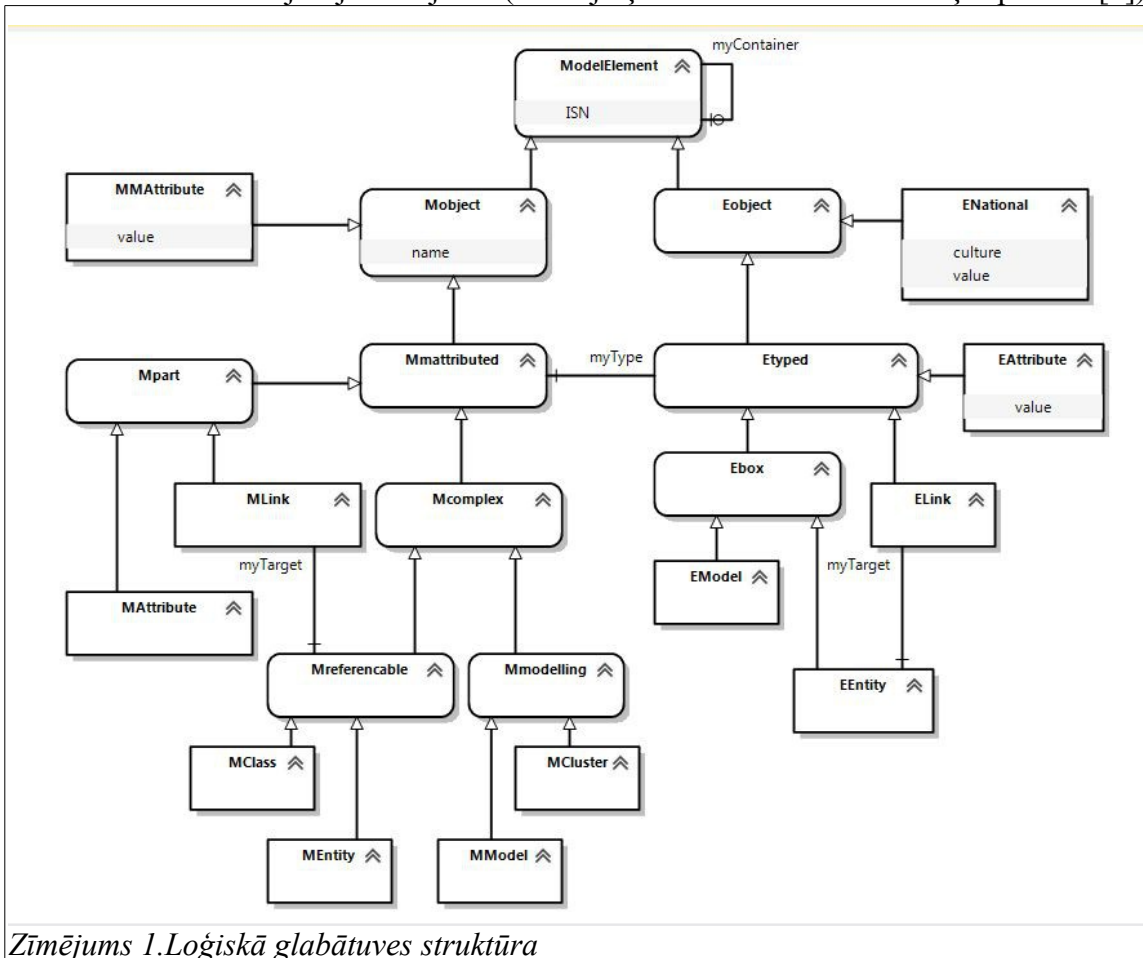
Šis pārskats ir veltīts projekta apakšaktivitātes Nr.1.3.1 "Navigācijas nodrošināšana" ietvaros veiktajiem pētījumiem. Pirmajā daļā tiek aprakstīts Navigācijas modelis, savukārt, dokumenta otrā daļā ir aprakstītas izstrādāto klašu navigācijas iespējas.

Aktivitātes ietvaros ir izstrādātas klases Model, ModelElement, Mattributed, Mpart, MAttributed, MLink, Mcomplex, Mreferencable, Etyped, Elink, kas atbilst loģiskās glabātuves struktūrai Zīmējumā 1.

3. Navigācijas modelis

Navigācija darbojas starp objektiem, kas atrodas atmiņā. Tādēļ sākumā aplūkosim objektu saistības modeli, kas kalpos kā skelets navigācijai.

Apstrādes laikā atmiņā atrodas gan metamodeļa gan modeļa objekti, kuru vissparīgā struktūra attēlota sekojošajā zīmējumā (notācija ņemta no meta metamodeļa apraksta [1]).



Zīmējums 1. Loģiskā glabātuves struktūra

Kreisā puse (Mobject) patiesībā saturīgi atkārtoti meta metamodeļi un tiek izmantota metamodeļa glabāšanai. Savukārt, labā puse (Eobject) satur modeļa informāciju.

Ļoti svarīgs atribūts ir *ISN*, kas viennozīmīgi identificē katru objektu. Navigācijas pamatprincips būs:

1. iegūt meklējamā objekta *ISN*,
2. pēc norādītā *ISN* izvilkt objektu no glabātuves.

Tehniski *ISN* kodēšanai tiek izmantots C# datu tips *int*. Konkrētu objektu *ISN* vērtības ir pozitīvi (>0) skaitļi.

Atsauces parasti ir *int* tipa un ir uzbūvētas pēc sekojošiem principiem:

- atsauce >0 ir mērķa objekta *ISN*
- atsauce == 0 mērķa objekta nav
- atsauce < 0 pašlaik netiek izmantots

Saturīgi navigācija atkarībā no izejas objekta sadalās 3 samērā dažādos veidos:

1. datu glabātuves navigācija
2. modeļa objekta navigācija
3. metamodeļa objekta navigācija

Navigācijas funkcijas izstrādātas pēc principa, ka tām vienmēr tiek padota norāde uz datu glabātuves (klase *Model*) eksemplāru. Tālākajās nodaļās aplūkosim navigācijas iespējas visās klasēs. Tā kā datu glabātuve ir tik centrāla, tad tās navigāciju aplūkosim kā pirmo.

4. Navigācijas funkcijas

4.1. Model (datu glabātuve)

4.1.1. ModelElement get(int key)

Atrast elementu pēc tā *ISN*. Parametri:

Nosaukums	Tips	Apraksts
Key	Int	Meklējamā objekta <i>ISN</i> .

Ja objekts ar norādīto atslēgu (*key*) nav atrodams glabātvē, tad tiek iniciēta avārija (Exception).

4.1.2. ModelElement this[int key]

Šī metode ir alternatīva sintaktiskā forma metodei *get* (4.1).

4.1.3. IEnumerable<ModelElement> parts(int dadISN)

Atrod visus glabātuves objektus, kuru *myContainer* ir *dadISN*. Alternatīvi var teikt, ka metode atrod visus “bērnus” objektam ar norādīto *ISN*. Parametri:

Nosaukums	Tips	Apraksts
dadISN	Int	Saturošā objekta <i>ISN</i> .

Ja objekts ar norādīto atslēgu (*dadISN*) nav atrodams glabātvē, tad tiek iniciēta avārija (Exception).

4.1.4. IEnumerable<T> tPartsQ<T>(int dadISN)

Ir iepriekš definētās *parts* (4.1.3) modifikācija - no visiem “bērniem” tiek atlasīti tikai tie ar norādīto datu tipu. Parametri:

Nosaukums	Tips	Apraksts
T	T: ModelElement	Meklējamā objekta tips.

4.1.5. List<T> tParts<T>(int dadISN)

Atšķirās no iepriekš aprakstītās *tPartsQ* (4.1.4) tikai ar rezultāta tipu (sarakstu ir tehniski ērtāk apstrādāt nekā abstraktāko *IEnumerable*). Parametri:

Nosaukums	Tips	Apraksts
T	T: ModelElement	Meklējamā objekta tips.

4.1.6. **MModel** mmodel

Ar šī pieprasījuma (C# property) palīdzību iegūst katrā glabātavē esošo vienīgo meta modeļa objektu.

4.1.7. **List<MEntity>** MEntityS

Ar šī pieprasījuma (C# property) palīdzību iegūst visas meta modelī definētās meta entītiņu instances.

4.1.8. **List<MClass>** MClassS

Ar šī pieprasījuma (C# property) palīdzību iegūst visas meta modelī definētās meta klašu instances.

4.1.9. **List<Etyped>** parts(int dadISN, int partTypeISN)

Ar šī pieprasījuma palīdzību modelī atrod kādam objektam pakārtotos objektus (tos objektus, kuru *myContainer* sakrīt ar norādīto). Parametri:

Nosaukums	Tips	Apraksts
dadISN	Int	tiek atlasīti objekti, kuru <i>myContainer</i> == <i>dadISN</i> .
partTypeISN	Int	tipa filtrs - tiek atlasīti tie objekti, kuru <i>myType</i> == <i>partTypeISN</i> .

4.1.10. **List<Etyped>** parts(int dadISN, List<int> partTypeISNs)

Paplašināta versija iepriekšējai funkcijai (4.1.9) - objekta tips ir kāds no saraksta.

4.1.11. **IEnumerable<T>** typed<T>(IEnumerable<ModelElement> from)

Šī funkcija atlasa objektus ar norādīto tipu no apstrādājamās objektu kopas. Apstrādājamā objektu kopa ir visi datu glabātavē objekti, ja *from* == *null* (noklusētā parametra vērtība), pretējā gadījumā - *from* saturs.

4.1.12. **string** hardcodedBase

Ar šī pieprasījuma (C# property) palīdzību iegūst bāzes metamodeļa - tā, no kura atvasināts šis metamodelis - vārdu.

4.2. ModelElement

4.2.1. `int myContainer`

Ar šī pieprasījuma (C# property) palīdzību iegūst saturošā objekta ISN (skat atsauču tipisko mehānismu 3).

4.2.2. `List<T> getParts<T>(Model store)`

Atrod šī objekta “bērnus” (tos objektus, kuru `myContainer`==šī objekta ISN). Parametri:

Nosaukums	Tips	Apraksts
T	T:ModelElement	Meklējamo objektu tips.
Store	Model	Apstrādājamā datu krātuve.

4.3. Mattributed

4.3.1. `IEnumerable<MMAtribute> MMAtributeS(Model store)`

Atrod metaobjekta visus meta metaatribūtus (tos, kuriem norādīta vērtība). Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.

4.3.2. `List<MMAtribute> MMAtributeM(Model store, string name)`

Atrod metaobjekta visus meta metaatribūtus ar norādīto vārdu, t.i. daudzvērtīgā meta metaatribūta visas vērtības. Tipiskās situācijas būs ar 1 elementu - ir meta metaatribūts ar norādīto vērtību un 0 elementu - meta metaatribūtam vērtības nav piešķirtas. Tātad šo metodi var pielietot ar jebkuru iespējamo meta metaatribūta vārdu. Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.
Name	string	Meta metaatribūta vārds.

4.3.3. `MMAtribute MMAtributeF(Model store, string name)`

Atrod metaobjekta pirmo meta metaatribūtu ar norādīto vārdu. Ja meta metaatribūtam vērtība nav piešķirta, tad rezultāts ir `null`. Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.
Name	string	Meta metaatribūta vārds.

4.3.4. string MMAtributeV(Model store, string name)

Atrod metaobjekta meta metaatribūta ar norādīto vārdu pirmo simbolisko vērtību. Ja meta metaatribūtam vērtība nav piešķirta, tad rezultāts ir *null*. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.
name	string	Meta metaatribūta vārds.

4.3.5. bool MMAtributeB(Model store, string name)

Atrod metaobjekta meta metaatribūta ar norādīto vārdu pirmo vērtību interpretētu kā loģisko vērtību, t.i. vērtība ir *true* tad un tikai tad, ja simboliskā vērtība (skat. 4.3.4) ir "true". Ja meta metaatribūtam vērtība nav piešķirta, tad rezultāts ir *false*. Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.
name	string	Meta metaatribūta vārds.

4.3.6. bool isTechnical(Model store)

Atrod metaobjekta meta metaatribūta "isTechnical" loģisko vērtību (skat. 4.3.5).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.4. Mpart

4.4.1. bool isMultiple(Model store)

Atrod metaobjekta meta metaatribūta "isMultiple" loģisko vērtību (skat. 4.3.5).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.4.2. bool isMandatory(Model store)

Atrod metaobjekta meta metaatribūta "isMandatory" loģisko vērtību (skat. 4.3.5).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.5. MAttribute (metaatribūts)

4.5.1. string dataType(Model store)

Atrod metaobjekta meta metaatribūta "dataType" simbolisko vērtību (skat. 4.3.4).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.5.2. public bool isName(Model store)

Atrod metaobjekta meta metaatribūta "isName" loģisko vērtību (skat. 4.3.5). Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.5.3. bool isNational(Model store)

Atrod metaobjekta meta metaatribūta "isNational" loģisko vērtību (skat. 4.3.5).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.5.4. string constraint(Model store)

Atrod metaobjekta meta metaatribūta "constraint" simbolisko vērtību (skat. 4.3.4).

Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.6. MLink (metanorāde)

4.6.1. int myTarget

Ar šī pieprasījuma (C# property) palīdzību iegūst *ISN* objektam (skat atsauču tipisko mehānismu 3), uz kuru norāda šī metanorāde (atbilstoši meta metamodelim tam jābūt *Mreferencable* eksemplāram).

4.6.2. Mreferencable myTargetO(Model store)

Atrod referencēto metaobjektu, t.i. to metaobjektu, uz kuru norāda *myTarget* (skat. 4.6.1). Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.7. Mcomplex

4.7.1. `int mySuperClass`

Ar šī pieprasījuma (C# property) palīdzību iegūst *ISN* objektam (skat atsauču tipisko mehānismu 3), kas ir šī metobjekta virsklase C# nozīmē (atbilstoši meta metamodelim tam jābūt *MClass* eksemplāram).

4.7.2. `List<int> myExtraSuperClasses`

Ar šī pieprasījuma (C# property) palīdzību iegūst *ISN* objektiem (skat atsauču tipisko mehānismu 3), kas ir šī metobjekta papildus virsklases (atbilstoši meta metamodelim tam jābūt *MClass* eksemplāram). Papildus virsklases tiek emulētas, jo C# nepieļauj daudzkāāršo mantošanu, un līdz ar to tiek nodrošināta tikai *Mpart* mantošana datu nozīmē (pilna C# mantošana atribūtiem un metodēm netiek nodrošināta).

4.7.3. `List<int> family(Model store, FamilyKind kind)`

Atrod metaobjekta virsklašu (skat. 4.7.1 un 4.7.2) *ISN* sarakstu (tai skaitā arī tranzitīvās virsklases). Šajā sarakstā tiek ielikts arī aplūkojamā metaobjekta *ISN*. Meklēšana var būt pilna vai tikai papildus virsklasēs. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.
kind	FamilyKind	Meklēšanas veids, Noklusētā vērtība <i>FamilyKind.full</i> - pilna meklēšana.

4.7.4. `List<T> metaPartS<T>(Model store, FamilyKind)`

Funkcija atrod norādītā tipa (*T*) meta sastāvdaļas (metaatribūti vai metanorādes) gan šim metaobjektam gan tā bāzes objektiem (skat.4.7.3), t.i. metasastāvdaļu meklēšana, ņemot vērā mantošanu. Parametri:

Nosaukums	Tips	Apraksts
T	T:Mpart	Meklējamā objekta tips.
store	Model	Apstrādājamā datu krātuve.
kind	FamilyKind	Meklēšanas veids, Noklusētā vērtība <i>FamilyKind.full</i> - pilna meklēšana.

4.7.5. `List<MAttribute> MAttributeS(MEDUS.Model store)`

Funkcija atrod visus metaobjekta tiešos (bez mantošanas) metaatribūtus. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.7.6. List<MAttribute> MAttributeSs(Model store)

Funkcija atrod visus metaobjekta metaatribūtus (mantošana atbilstoši 4.7.4). Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.

4.7.7. List<MLink> MLinkS(Model store)

Funkcija atrod visus metaobjekta tiešās (bez mantošanas) metanorādes. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.7.8. List<MLink> MLinkS(Model store)

Funkcija atrod visus metaobjekta metanorādes (mantošana atbilstoši 4.7.4). Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.7.9. int myNameAttribute(Model store)

Funkcija atrod šī metaobjekta metaatribūta, kurš ir atzīmēts ar *isName==true*, ISN. Ja šāds metaatribūts netiek atrasts, tad rezultāts ir vērtība 0. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.7.10. bool isComplexVisibleISN(Model store)

Atrod metaobjekta meta metaatribūta "*isExtendedVisibleISN*" loģisko vērtību (skat. 4.3.5). Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.8. Mreferencable

4.8.1. List<int> eTypes(Model store)

Atrod referencējama metaobjekta bāzes meta klases (saraksts ietver arī aplūkojamo objektu, kas *MEntity* gadījumā ir vienīgais rezultāta elements) kā *ISN* sarakstu. Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.9. Etyped

4.9.1. **int** myType

Ar šī pieprasījuma (C# property) palīdzību iegūst *ISN* metaobjektam (skat atsauču tipisko mehānismu 3), uz kuru norāda šī metanorāde (atbilstoši meta metamodelim tam jābūt *Mmattributed* eksemplāram).

4.9.2. **Mmattributed** myTypeO(**Model** store)

Atrod referencēto metaobjektu, t.i. to metaobjektu, uz kuru norāda *myType* (skat. 4.9.1).
Parametri:

Nosaukums	Tips	Apraksts
store	Model	Apstrādājamā datu krātuve.

4.9.3. **string** myTypeN(**Model** store)

Atrod aplūkojamā modeļa objekta tipu jeb tehniski tā metaobjektu, uz kuru norāda *myType* (skat. 4.9.1) atribūta *name* vērtību. Parametri:

Nosaukums	Tips	Apraksts
Store	Model	Apstrādājamā datu krātuve.

4.10. ELink (modeļa norāde)

4.10.1. **int** myTarget

Ar šī pieprasījuma (C# property) palīdzību iegūst *ISN* referencētajam modeļa objektam (skat atsauču tipisko mehānismu 3).

5. Literatūras saraksts

[1] Projekta aktivitātes Nr.1.1 "Meta metamodeļu izpēte" progresu pārskats.