



**ΕΛΛΗΝΙΚΟ ΑΝΟΙΚΤΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ**

ΠΡΟΓΡΑΜΜΑ ΣΠΟΥΔΩΝ

**ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΞΕΙΔΙΚΕΥΣΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ
ΣΥΣΤΗΜΑΤΑ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΙ ΜΟΝΤΕΛΩΝ.ΑΠΟ ΤΗΝ
ΑΝΑΛΥΣΗ ΠΡΟΔΙΑΓΡΑΦΩΝ ΣΤΗΝ ΑΝΑΛΥΣΗ ΚΑΙ ΤΗΝ
ΣΧΕΔΙΑΣΗ ΕΝΟΣ ΣΥΣΤΗΜΑΤΟΣ**

ΑΡΓΥΡΙΟΥ ΑΝΤΩΝΙΟΣ

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΕΠΙΒΛΕΠΟΝΤΟΣ ΚΑΘΗΓΗΤΗ
κος ΜΠΕΛΗΓΙΑΝΝΗΣ ΓΡΗΓΟΡΙΟΣ**

**ΠΑΤΡΑ
ΣΕΠΤΕΜΒΡΙΟΣ 2010**



*Αργυρίου Αντώνιος, ΈΜετασχηματισμοί μοντέλων : Από την ανάλυση
προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'*

Διπλωματική Εργασία

Μετασχηματισμοί μοντέλων.Απο την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος

Αργυρίου Αντώνιος

Σεπτέμβριος 2010



*Αργυρίου Αντώνιος, ΈΜετασχηματισμοί μοντέλων : Από την ανάλυση
προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'*

ΕΑΠ, 2010

Η παρούσα διατριβή, η οποία εκπονήθηκε στα πλαίσια της ΘΕ «Διπλωματική Εργασία» του προγράμματος «Μεταπτυχιακή Εξειδίκευση στα Πληροφοριακά Συστήματα» (ΠΛΗΣ), και τα λοιπά αποτελέσματα της αντίστοιχης Διπλωματικής Εργασίας (ΠΕ) αποτελούν συνιδιοκτησία του ΕΑΠ και του φοιτητή, ο καθένας από τους οποίους έχει το δικαίωμα ανεξάρτητης χρήσης και αναπαραγωγής τους (στο σύνολο ή τμηματικά) για διδακτικούς και ερευνητικούς σκοπούς, σε κάθε περίπτωση αναφέροντας τον τίτλο και το συγγραφέα και το ΕΑΠ, όπου εκπονήθηκε η Διπλωματική Εργασία, καθώς και τον επιβλέποντα και την επιτροπή κρίσης.



Αργυρίου Αντώνιος, ΈΜετασχηματισμοί μοντέλων : Από την ανάλυση
προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος

Αργυρίου Αντώνιος

Ονοματεπώνυμο
Επιβλέποντα

Ονοματεπώνυμο
Μέλους 1

Ονοματεπώνυμο
Μέλους 2

Μπεληγιάννης Γρηγόριος

Χατζευθυμιάδης
Ευστάθιος

Τσαγκαράκης Μανώλης

Περίληψη

Μια μοντέρνα κατεύθυνση της τεχνολογίας λογισμικού είναι αυτή του μετασχηματισμού μοντέλων (model2model transformation ή M2M transformation). Αυτή η τεχνική χρησιμοποιείται στην Οδηγούμενη από Μοντέλα Ανάπτυξη Λογισμικού (Model Driven Development) και την αρχιτεκτονική MDA (Model Driven Architecture). Σύμφωνα με την τεχνολογία αυτή, η μηχανική λογισμικού μπορεί να ωφεληθεί πολύ από την μεταφορά της πληροφορίας που υπάρχει σε μοντέλα μιας φάσης σε μοντέλα της επόμενης φάσης..

Έτσι ένα μοντέλο της φάσης ανάλυσης συστημάτων μπορεί να μετατραπεί σε μοντέλα της φάσης σχεδίασης λογισμικού. Το αρχικό μοντέλο λέγεται και μοντέλο πηγής (source model), ενώ το μοντέλο που δημιουργείται λέγεται μοντέλο στόχος (target model). Αυτή η διαδικασία γίνεται από μια γλώσσα μετασχηματισμού (με πιο γνωστή την ATL-Atlas Transformation Language) που έχει σαν είσοδο το μοντέλο πηγής και τα μεταμοντέλα πηγής και στόχου και δημιουργούν στην έξοδο το μοντέλο στόχο. Μεταμοντέλα είναι τα μοντέλα των μοντέλων δηλ. περιγράφουν πώς μπορούμε να δημιουργήσουμε ένα αρχείο συγκεκριμένου τύπου π.χ. εν προκειμένω ένα αρχείο XLM. Μια προσέγγιση αυτού του



μετασχηματισμού μπορεί να γίνει με χρήση εργαλείων γραφικής αναπαράστασης μοντέλων και μεταμοντέλων , όπως είναι τα EMF (Eclipse Modeling Framework) , GMF (Graphics Modeling Framework) και η πλατφόρμα ανάπτυξης λογισμικού Eclipse.

Στα πλαίσια της παρούσας διπλωματικής, μελετήθηκαν οι MDE τεχνολογίες δημιουργίας μοντέλων και μετα-μοντέλων και μετασχηματισμού μοντέλων , η μεθοδολογία δημιουργίας πρακτοροστραφών συστημάτων (ASEME) και τα μοντέλα της Γλώσσας Μοντελοποίησης Πρακτόρων-AMOLA(Agent Modeling Language).

Επίσης για τις ανάγκες της εργασίας , κατασκευάστηκαν νέα μεταμοντέλα με βάση διαθέσιμα στην επιστημονική κοινότητα με χρήση τεχνολογίας EMF και έγινε γραφική αναπαράσταση των μοντέλων ASEME των παραπάνω μεταμοντέλων , με χρήση εργαλείων τεχνολογίας GMF , με απώτερο στόχο την πιο παραστατική άποψη του μετασχηματισμού των εν λόγω μοντέλων.

Λέξεις-κλειδιά: Μετασχηματισμοί μοντέλων (model2model transformation-M2M transformation) , μεταμοντέλα , Μεθοδολογία Δημιουργίας Πρακτοροστραφών Συστημάτων ASEME(Agent Systems Engineering Methology) , Γλώσσα Μοντελοποίησης Πρακτόρων AMOLA(Agent Modeling Language) , Μηχανική Οδηγούμενη από Μοντέλα MDE (Model Driven Engineering) , Επεκτάσιμη Γλώσσα σήμανσης XML(eXtensible Markup Language) , Τεχνολογίες EMF(Eclipse Modeling Framework) και GMF(Graphics Modeling Framework).

Περιεχόμενο: Κείμενο , οθόνες HY , διαγράμματα μοντέλων και μεταμοντέλων , λίστες αρχείων σε xml μορφή.



*Αργυρίου Αντώνιος, Έμετασχηματισμοί μοντέλων : Από την ανάλυση
προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'*

**Model transformations.From the requirements analysis to the system analysis and
design.**

Argyriou Antonios

Supervisor

Co-examiner

Co-examiner

Dr. Hadjiefthymiadis

Dr. Beligiannis Grigorios

Efstathios

Dr. Tsagarakis Manolis

Abstract

A Modern direction of software technology is that of a model transformation (model2model transformation or M2M transformation).This technique is used both in Model Driven Development and MDA (Model Driven Architecture).According to this technology , Software Engineering could benefit to the highest degree from the information transfer which exists in one-phase models to models of the next phase.

In this way a model of system analysis phase can be turned into models of a software design phase.The initial model is also called the source model , whereas the created model is known as the target model.This process is done through a transformation language (ATL-Atlas Transformation Language being the most popular one) which uses the source model , both the source and target metamodels as an input and as a result the output is the target model.Metamodels are the models of the models which means that they describe how we can create a file of a particular type , e.g. in this case an XML file.An approach of this transformation can be accomplished by the use of model and metamodel graphic presentation , such as EMF (Eclipse Modeling Framework) and the development platform of eclipse software.

As far as this thesis , is concerned MDE technology for the creation of models and metamodels , model transformation , the Agent Systems Engineering methodology (ASEME) and the models of the Agent Modeling Language (AMOLA) have been studied.

Futhermore , due to the demand of this thesis , new metamodels have been created , based on available ones in the scientific community with the use of EMF technology and a graphic presentation of ASEME models of the metamodels , has also been created , with the use of GMF technology tools , targeting on a more emblematic viewpoint of the models – in question – transformation.

Keywords: Model2model transformation-M2M transformation), ASEME (Agent Systems Engineering Methology) , AMOLA (Agent Modeling Language) , MDE (Model Driven Engineering) , XML(eXtensible Markup Language) , EMF (Eclipse Modeling Framework) , GMF (Graphics Modeling Framework).

Content : text , screenshots , models and metemodells diagrams , file listings(xml format)



Ευρετήριο εικόνων

Εικ.1	Παρουσίαση του σχήματος μεταμοντελοποίησης (A) με ορισμό μοντέλου (B).....	20
Εικ.2	Το EMF μοντέλο ενοποιεί Java , XML και UML τεχνολογίες	21
Εικ.3	Το ecore μεταμοντέλο	21
Εικ.4	Το γενικό σχήμα του μοντέλου μετασχηματισμού	24
Εικ.5	Η πορεία της εκτέλεσης του ATL μετασχηματισμού	29
Εικ.6	Οι φάσεις της μεθοδολογίας ASEME και τα AMOLA προϊόντα τους	31
Εικ.7	Η φάση ανάλυσης απαιτήσεων της ASEME	31
Εικ.8	Η φάση ανάλυσης συστήματος της ASEME	32
Εικ.9	Η φάση σχεδίασης συστήματος της ASEME	32
Εικ.10	Το AMOLA SAG μεταμοντέλο	35
Εικ.11	Το AMOLA SUC μεταμοντέλο	35
Εικ.12	Το AMOLA SRM μεταμοντέλο	36
Εικ.13	Το AMOLA IAC μεταμοντέλο	37
Εικ.14	Η δομή ενός project ATL μετασχηματισμού στο Ολοκληρωμένο Περιβάλλον Ανάπτυξης Eclipse (IDE Eclipse)	39
Εικ.15	Το σχήμα του SUC2SRM M2M μετασχηματισμού	40
Εικ.16	Διάγραμμα δραστών για ένα media market	43
Εικ.17	Η 5 φάσεων διαδικασία της Tropos και το περιβάλλον ανάπτυξης οδηγούμενο από μοντέλα που υποστηρίζει την μεθοδολογία	47
Εικ.18	Στοιχεία της άποψης πράκτορα στην IGNENIAS	48
Εικ.19	Επισκόπηση του πλαισίου GMF	52
Εικ.20	Δημιουργία Ecore μεταμοντέλου	52
Εικ.21	Δημιουργία Ecore μεταμοντέλου	53
Εικ.22	Καθορισμός σχεδίασης με το GMF (Tooling Definition)	53
Εικ.23	Καθορισμός χαρτογράφησης με το GMF (Mapping Definition)	54
Εικ.24	Τα gmfgraph , gmftool & gmfmap αρχεία υπό δένδρική μορφή	54
Εικ.25	Η επιφάνεια σχεδίασης του τελικά δημιουργούμενα γραφικού συντάκτη	55
Εικ.26	Το 'annotated' μεταμοντέλο Filesystem	56
Εικ.27	Παράδειγμα GMF συντάκτη που δημιουργήθηκε με το EuGENia ..	57
Εικ.28	Ταμπλό του Eclipse EMF-GMF (dashboard)	58
Εικ.29	Το νέο SAG μεταμοντέλο (ecore_diagram)	61
Εικ.30	Το νέο SAG μεταμοντέλο στην εκτεταμένη του μορφή	62
Εικ.31	Το νέο SAG Generator model και η όψη των ιδιοτήτων του	63
Εικ.32	Το νέο SUC μεταμοντέλο (ecore_diagram)	65
Εικ.33	Το νέο SUC μεταμοντέλο στην εκτεταμένη του μορφή	65
Εικ.34	Το νέο AIP μεταμοντέλο (ecore_diagram)	67
Εικ.35	Το νέο AIP μεταμοντέλο στην εκτεταμένη του μορφή	68
Εικ.36	Το νέο SRM μεταμοντέλο (ecore_diagram)	69
Εικ.37	Το νέο SRM μεταμοντέλο στην εκτεταμένη του μορφή	70
Εικ.38	Το νέο IAC μεταμοντέλο (ecore_diagram)	72



Εικ.39	Το νέο IAC μεταμοντέλο στην εκτεταμένη του μορφή	72
Εικ.40	To SAG Graphical Definition Model	75
Εικ.41	To SAG Tooling Definition Model	76
Εικ.42	To SAG Mapping Definition Model	77
Εικ.43	Το νέο SAG μοντέλο (όνομα αρχείου SAG_sample.sagd)	78
Εικ.44	Το νέο Domain SAG μοντέλο (όνομα αρχείου SAG_sample.sag)	79
Εικ.45	To SUC Graphical Definition Model	80
Εικ.46	To SUC Tooling Definition Model	81
Εικ.47	To SUC Mapping Definition Model	82
Εικ.48	Το νέο SUC μοντέλο (όνομα αρχείου SUC_sample.sucd)	83
Εικ.49	Το νέο Domain SUC μοντέλο (όνομα αρχείου SUC_sample.suc)	84
Εικ.50	To AIP Graphical Definition Model	85
Εικ.51	To AIP Tooling Definition Model	86
Εικ.52	To AIP Mapping Definition Model	86
Εικ.53	Το νέο AIP μοντέλο (όνομα αρχείου AIP_sample.aipd)	88
Εικ.54	Το νέο Domain AIP μοντέλο (όνομα αρχείου AIP_sample.aip)	88
Εικ.55	To SRM Graphical Definition Model	89
Εικ.56	To SRM Tooling Definition Model	90
Εικ.57	To SRM Mapping Definition Model	91
Εικ.58	Το νέο SRM μοντέλο (όνομα αρχείου SRM_sample.srmd)	92
Εικ.59	Το νέο Domain SRM μοντέλο (όνομα αρχείου SRM_sample.srm)	93
Εικ.60	To IAC Graphical Definition Model	94
Εικ.61	To IAC Tooling Definition Model	95
Εικ.62	To IAC Mapping Definition Model	96
Εικ.63	Το νέο IAC μοντέλο (όνομα αρχείου IAC_sample.iacd)	
	Δενδρική αναπαράσταση του διαγράμματος καταστάσεων	97
Εικ.64	Το νέο Domain IAC μοντέλο (όνομα αρχείου IAC_sample.iac)	98
Εικ.65	Το αρχείο του μετασχηματισμού SAG2SUC(SAG2SUC.atl)	99
Εικ.66	Run Configuration του ATL μετασχηματισμού SAG2SUC του Eclipse	100
Εικ.67	Το αρχείο του μετασχηματισμού SUC+AIP2SRM(SUC_AIP2SRM.atl)	101
Εικ.68	Run Configuration του ATL μετασχηματισμού (SUC+AIP2SRM) του Eclipse	102
Εικ.69	Δομή της πλατφόρμας eclipse	115
Εικ.70	Αρχική μορφή του πάγκου εργασίας του eclipse	116
Εικ.71	Όψη Outline για ένα αρχείο πηγαίου κώδικα Java	12-
Εικ.72	Δημιουργία ενός νέου plug-in με τη βοήθεια του PDE	121
Εικ.73	Αρχιτεκτονικό μόρφημα MVC	132
Εικ.74	Ένα δέντρο σχημάτων και η γραφική αναπαράστασή του αν κάθε σχήμα ζωγραφιστεί ως πλήρες ορθογώνιο παραλληλόγραμμο	134
Εικ.75	Η αρχιτεκτονική MVC όπως υλοποιείται από το GEF	137
Εικ.76	Δενδρική δομή των EditParts	138



Αρκτικόλεξα-Συντομογραφίες

AIP	Agent Interaction Protocols model
AMMA	ATLAS Model Management Architecture
AMOLA	Agent Modelling Language
AOSE	Oriented Software Engineering
API	Application Programming Interface
ASEME	Agent Systems Engineering Methology
ATLAS	Atlas Transformation Language
CORBA	Common Object Request Broker Architecture
DSL	Domain Specific Language
DTD	Document Type Definition
EBNF	Extended Backus- Naur Form
EMF	Eclipse Modeling Framework
GEF	Graphical Editing Framework
GMF	Graphical Modeling Framework
GPL	General Public License
GUI	Graphical Unit Interface
HTML	HyperText Markup Language
IAC	Intra-Agent Control Model
IDK	Ignenias Development Kit
JADE	Java Agent Development Framework
MAS	Multi-Agent Systems
MDE	Model Driven Engineering
MDT	Model Development tools
MOF	Meta-Object Facility
MVC	Model-View-Controller
M2M transformation	Model to Model transformation
M2T transformation	Model to Text transformation
OCL	Object Constraint Language



*Αργυρίου Αντώνιος, Έμετασχηματισμοί μοντέλων : Από την ανάλυση
προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'*

PCDATA	Parsed Character Data
PDE	Plug-in Development Environment
PIM	Platform Independent Model
PSM	Platform Specific Model
QVT	Query-View -Transformation
RCP	Platform Rich Client
RPG	Requirements Per Goal
SAG	System Actors Goals
SDO	Service Data Objects
SPEM	Software Process Engineering Metamodel
SRM	System Roles Model
SUC	System Use Cases
T2M transformation	Text to Model transformation
T2T transformation	Text to Text transformation
TAOM4E	Tool for Agent Oriented Modeling for Eclipse
TGG	Triple Graph Grammars
UML	Unified Modeling Language
XSD	XML Schema Definition



Περιεχόμενα

Κεφάλαιο 1 : Εισαγωγή	14
1.1 Στόχοι της εργασίας	15
1.2 Διάρθρωση της εργασίας	16
Κεφάλαιο 2 : Θεωρητικό υπόβαθρο	18
2.1 Μηχανική Οδηγούμενη από Μοντέλα (Model Driven Engineering-MDE)...	20
2.1.1 Μετα- και μεταμετα – μοντέλα	20
2.1.2 Μετασχηματισμοί μοντέλων	23
2.2 Μεθοδολογία Ανάπτυξης Πρακτοροστροφών Συστημάτων (Agent Systems Engineering Methodology-ASEME)...	30
2.2.1 Μεταμοντέλα	33
2.2.2 Μετασχηματισμοί μοντέλων	38
2.3 Χρήση γραφικών εργαλείων σε Υπάρχουσες Μεθοδολογίες	40
2.3.1 Μεθοδολογία Tropos	42
2.3.2 Μεθοδολογία Ignenias	46
2.4 Εργαλεία Γραφικής Αναπαράστασης Μοντέλων	49
2.4.1 Graphical Modeling Framework (GMF)- Eclipse Modeling Framework (GMF)	49
2.4.2 Eclipse Epsilon - EuGENia	55
Κεφάλαιο 3: Γραφική αναπαράσταση μοντέλων Μεθοδολογίας Ανάπτυξης Πρακτοροστροφών Συστημάτων (Agent Systems Engineering Methodology-ASEME)...	58
3.1 Τα νέα EMF μεταμοντέλα	60
3.1.1 Μεταμοντέλο SAG	60
3.1.2 Μεταμοντέλο SUC	62
3.1.3 Μεταμοντέλο AIP	66
3.1.4 Μεταμοντέλο SRM	68
3.1.5 Μεταμοντέλο IAC	71
3.2 Τα γραφικά εργαλεία	73



3.2.1 Γραφικό μοντέλο επεξεργασίας SAG διαγραμμάτων	74
3.2.2 Γραφικό μοντέλο επεξεργασίας SUC διαγραμμάτων	79
3.2.3 Γραφικό μοντέλο επεξεργασίας AIP διαγραμμάτων	84
3.2.4 Γραφικό μοντέλο επεξεργασίας SRM διαγραμμάτων.....	88
3.2.5 Γραφικό μοντέλο επεξεργασίας IAC διαγραμμάτων	93
Κεφάλαιο 4 : Μετασχηματισμοί στην ASEME	99
4.1 Μετασχηματισμός SAG2SUC	99
4.2 Μετασχηματισμός SUC+AIP2SRM	101
Κεφάλαιο 5 : Συμπεράσματα – Αξιολογήσεις- Μελλοντικές κατευθύνσεις	103
5.1 Μελλοντικές κατευθύνσεις της έρευνας	105
Αναφορές	108
Παράρτημα Α : Η πλατφόρμα του Eclipse	113
Παράρτημα Β : Το πλαίσιο Graphical Editing Framework	131
Παράρτημα Γ : Η γλώσσα XML	142
Παράρτημα Δ : Τα αρχεία των μεταμοντέλων (ecore αρχεία- xmi)	146
Παράρτημα Ε : 'Το σύστημα διαχείρισης συναντήσεων' (αρχεία- xmi)	152



Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή Dr. Κο Γρηγόριο Μπεληγιάννη για την εμπιστοσύνη που μου έδειξε όταν του πρότεινα να επιβλέψει τη διπλωματική μου εργασία και για την διάθεσή του να μου λύσει όμι απορίες δημιουργούνταν στην πορεία της εκπόνησης της εργασίας μου.

Ευχαριστώ ιδιαίτερα και εκ βάθους ψυχής τον Δρ. Νικόλαο Σπανουδάκη , ΕΕΔΙΠ ΙΙ του Πολυτεχνείου Κρήτης , για την πολύ ουσιαστική και άοκνη συνεργασία μας, την αδιάκοπη επικοινωνία μας και κύρια για τις πολύωρες τηλεσυνεδρίες μας , που με την γρήγορη ανταπόκριση στα ερωτήματα/απορίες μου και την έγκυρη καθοδήγηση του, με συνέπεια και υπευθυνότητα καθ' όλη τη διάρκεια εκπόνησης της διπλωματικής μου εργασίας, μου έδειξε ένα διαφορετικό τρόπο σκέψης και θεώρησης και αποτέλεσαν πηγή έμπνευσης και τεχνογνωσίας. και με βοήθησε πολύ να επικεντρώνομαι σωστά στα επί μέρους σημεία της εργασίας , όντας βαθύς γνώστης- ερευνητής του συγκεκριμένου γνωστικού αντικειμένου.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένεια μου για την κατανόηση και την υπομονή της κατά το διάστημα υλοποίησης αυτής της διπλωματικής εργασίας.



1. Εισαγωγή

Η πρακτοροστραφής ανάπτυξη αναδύεται ως ένας μοντέρνος τρόπος δημιουργίας λογισμικού. Το κύριο πλεονέκτημά της είναι ότι κατορθώνει να αναπτύσσει ένα έξυπνο, κοινωνικό και αυτόνομο λογισμικό, τρεις ιδιότητες που το κάνουν διαφορετικό σε σχέση με το παραγόμενο από την κλασσική αντικειμενοστραφή σχεδίαση. Η ανάπτυξη ενός τέτοιου μοντέρνου λογισμικού δεν θα πρέπει μόνο να παίρνει υπ' όψιν τις ανάγκες του αναλυτή, αλλά και τις ανάγκες του σχεδιαστή του λογισμικού. Μια από τις ανάγκες αυτές είναι το παραγόμενο λογισμικό να περιλαμβάνει και να ενσωματώνει διαφορετικές και προηγμένες τεχνολογίες όπως προγραμματισμός, διαλεκτικός συλλογισμός, όραση υπολογιστών κλπ. Οι ανάγκες αυτές μπορούν να εξετασθούν από μια μεθοδολογία που παρέχει πολλά αφαιρετικά επίπεδα έτσι ώστε να μπορεί να μοντελοποιηθεί πολύπλοκο λογισμικό. Επιπλέον, η τεχνολογία αυτή θα πρέπει να δεικνύει τις αναγκαίες τεχνολογίες από τα στάδια του πρώιμου σχεδιασμού και επιτρέπει μια σαφή διάσπαση έργων. Επίσης η προσέγγιση του σχεδιασμού ενοτήτων θα οδηγήσει σε μια σαφή και ταχεία διεργασία σύνθεσης.

Ο αυτοματισμός είναι μια σπουδαία πτυχή της ανάπτυξης λογισμικού. Η Οδηγούμενη από Μοντέλα Μηχανική είναι ένα νέο παράδειγμα ανάπτυξης λογισμικού που προτείνει μια διαδικασία ανάπτυξης λογισμικού βασισμένη σε μετασχηματισμούς μοντέλων μεταξύ των διαφορετικών φάσεων ανάπτυξης. Η μοντέρνα μεθοδολογία θα πρέπει να λάβει υπόψη αυτή την τάση, καθώς εξυπηρετεί μη λειτουργικές απαιτήσεις όπως είναι η φορητότητα, η υπερλειτουργικότητα και η επαναχρησιμοποιησιμότητα.

Η Μεθοδολογία Ανάπτυξης Πρακτοροστραφών Συστημάτων (Agent Systems Engineering Methodology –ASEME) είναι μια μεθοδολογία για την δημιουργία τέτοιων συστημάτων. Άρχισε καθώς η Gaia2JADE διαδικασία για υλοποίηση Gaia μοντέλων [40] χρησιμοποίησε την JADE πλατφόρμα πρακτόρων [1] και αναδύθηκε ως εξέλιξη της διαδικασίας αυτής, επηρεασμένη από την φάση της ανάλυσης απαιτήσεων της μεθοδολογίας Tropos [4], της εργασίας του Moore στις πολιτικές συνομιλίας [27] και τις εργασίες στην μηχανική των συστημάτων πραγματικού κόσμου [22], [26], [24], [25], [33], [32], [34].



1.1 Στόχοι της εργασίας

Σκοπός της παρούσας Διπλωματικής Εργασίας είναι να αυτοματοποιήσει την μετατροπή ενός μοντέλου της φάσης ανάλυσης προδιαγραφών της ASEME μεθοδολογίας , που βασίζεται στο διάγραμμα δραστών (actor diagram) της μεθοδολογίας Tropos , σε ένα μοντέλο της φάσης ανάλυσης της ASEME μεθοδολογίας , που βασίζεται στα UML διαγράμματα περιπτώσεων χρήσης (usecases diagrams). Ακολούθως να αυτοματοποιήσει την μετατροπή του μοντέλου της φάσης ανάλυσης της ASEME μεθοδολογίας σε μοντέλο της φάσης σχεδίασης της εν λόγω μεθοδολογίας , που βασίζεται στα διαγράμματα καταστάσεων (statecharts). Τα διαγράμματα καταστάσεων , για λόγους ευκρινούς παρουσίασης , θα αναπαρασταθούν ως κλάδοι δένδρου (tree Branch) με χρήση υποδειγμάτων εκτεταμένων Gaia τελεστών.

Τα ανωτέρω μοντέλα , αλλά και τα αντίστοιχα μεταμοντέλα τους θα αναπαρασταθούν γραφικά με χρήση γραφικών εργαλείων ολοκληρωμένου περιβάλλοντος ανάπτυξης λογισμικού.

Επίσης , σκοπεύει να κάνει μια εν μέρει αποτίμηση, όσον αφορά των συνδυασμό των λύσεων και τεχνολογιών που υπάρχουν για την επίλυση αυτού του σύνθετου προβλήματος.

Αναλυτικότερα η εργασία αυτή έχει ως στόχους :

- την παρουσίαση της Οδηγούμενης από Μοντέλα Μηχανικής (Model Driven Engineering-MDE) , μιας τεχνικής που χρησιμοποιείται για τον μετασχηματισμό μοντέλων
- ειδικότερα , την παρουσίαση της Μεθοδολογίας Δημιουργίας Πρακτοροστραφών Συστημάτων (Agent Systems Engineering Methology-ASEME)
- την διερεύνηση των διαθέσιμων γλωσσών μετασχηματισμού και μέσα από τα χαρακτηριστικά τους , την αποτίμηση της δυνατότητά τους για την διενέργεια των μετασχηματισμών της παρούσας εργασίας
- την επιλογή κατάλληλων μεταμοντέλων πηγής και στόχου από τα διαθέσιμα στην επιστημονική κοινότητα



- την τροποποίηση των επιλεγέντων μεταμοντέλων και η εξ' αρχής δημιουργία νέου μεταμοντέλου για συγκεκριμένο τύπο μοντέλου (πρωτοκόλλου διαδραστικότητας πρακτόρων) για τις ανάγκες της εργασίας , όπως και η δημιουργία των αντίστοιχων μοντέλων πηγής και στόχου αυτών
- την υλοποίηση των παραπάνω αναφερθέντων μετασχηματισμών μοντέλων , με χρήση γραφικών εργαλείων μέσα από το ανοικτό ολοκληρωμένο περιβάλλον της πλατφόρμας του Eclipse.

1.2 Διάρθρωση εργασίας

Το κύριο αντικείμενο αυτής της εργασίας είναι η παρουσίαση της MDE μεθοδολογίας ASEME , η διαδικασία ανάπτυξης με τα βήματα και τα προϊόντα της , όπως και οι μετασχηματισμοί μοντέλων – που αναπαρίστανται με γραφικό τρόπο - μεταξύ των διαφορετικών φάσεων ανάπτυξης. Τα μοντέλα που χρησιμοποιήθηκαν από την ASEME ορίστηκαν από την Γλώσσα Μοντελοποίησης Πρακτόρων (Agent Modeling Language- AMOLA). Η εργασία αυτή περιέχει ένα παράδειγμα εργασίας , την ανάπτυξη του «συστήματος διαχείρισης συναντήσεων» (Spanoudakis , 2009) [33] , το οποίο επιτρέπει την κατανόηση της διαδικασίας της ASEME και των μοντέλων της AMOLA.

Στο 1^ο Κεφάλαιο γίνεται μια εισαγωγή στο πεδίο και το αντικείμενο με το οποίο ασχολείται η διπλωματική εργασία , αναφέρεται ο σκοπός και οι επί μέρους στόχοι της καθώς και η διάρθρωσή της

Στο 2^ο Κεφάλαιο παρουσιάζεται το θεωρητικό υπόβαθρο του αντικειμένου της εργασίας , με παρουσίαση των χαρακτηριστικών της Οδηγούμενης από Μοντέλα Μηχανικής-MDE , που χρησιμοποιεί την τεχνική του μετασχηματισμού μοντέλων καθώς και της μεθοδολογίας δημιουργίας πρακτοροστραφών συστημάτων λογισμικού , που αποτελεί μια ιδιαίτερη μεθοδολογία της MDE. Γίνεται αναφορά στα γραφικά εργαλεία που χρησιμοποιούν οι μεθοδολογίες αυτές για να παραστήσουν τα μεταμοντέλα και τα μοντέλα τους. Επίσης γίνεται μια παρουσίαση των κυριότερων διαθέσιμων γλωσσών μετασχηματισμού και μια αναφορά εκτίμησης των χαρακτηριστικών τους , μέσα από τα πλεονεκτήματα και τα μειονεκτήματά τους.



Στο 3^ο Κεφάλαιο γίνεται τροποποιημένη κατασκευή – με γραφική αναπαράσταση – των μεταμοντέλων της ASEME , για το παράδειγμα εργασίας «διαχείριση συναντήσεων» και για τις ανάγκες της παρούσας εργασίας.Επίσης δημιουργείται εξ αρχής το μεταμοντέλο του μοντέλου πρωτοκόλλων αλληλεπίδρασης πρακτόρων (Agent Interaction Protocols model-AIP) , που δεν υπήρχε αρχικά.Τέλος κατασκευάζονται τα γραφικά μοντέλα επεξεργασίας των αντίστοιχων μοντέλων – δηλ. οι γραφικοί συντάκτες, των SAG , SUC , AIP , SRM και IAC μοντέλων , με τα οποία δημιουργούμε τα παραδείγματα-διαγράμματα που είναι απαραίτητα για τους μετασχηματισμούς.

Στο 4^ο Κεφάλαιο δημιουργούμε τους νέους μετασχηματισμούς SAG2SUC και SUC+AIP2SRM , για τις ανάγκες της εργασίας αυτής.

Στο 5^ο Κεφάλαιο παρουσιάζονται τα συμπεράσματα από την εργασία και οι μελλοντικές κατευθύνσεις, τα ανοικτά θέματα του παρόντος επιστημονικού πεδίου του μετασχηματισμού μοντέλων.



2. Θεωρητικό Υπόβαθρο

Το συγκεκριμένο κεφάλαιο αφορά στην ανάδειξη των τεχνικών μεθοδολογιών και των μοντέλων , επί των οποίων βασίσθηκε η παρούσα διπλωματική εργασία, καθώς και στην συνοπτική παρουσίαση άλλων επιστημονικών έργων που σχετίζονται με το εν λόγω αντικείμενο ή με παραπλήσιες θεματικές περιοχές.

2.1 Μηχανική Οδηγούμενη από Μοντέλα (MDE)

Η MDE [3] είναι η συστηματική χρήση των μοντέλων ως πρωταρχικά εργαλεία καθ'όλο τον κύκλο ζωής του έργου λογισμικού. Είναι συμβατή με το πρόσφατο αναδύόμενο παράδειγμα της Αρχιτεκτονικής Οδηγούμενης από Μοντέλα (Model Driven Architecture-MDA) [20]. Το δυνατό σημείο της MDA είναι ότι στοχεύει σε φορητότητα , διαλειτουργικότητα και επαναχρησιμότητα , τρεις μη-λειτουργικές απαιτήσεις που θεωρούνται ως πολύ σπουδαίες για τον σχεδιασμό μοντέρνων συστημάτων. Η MDA ορίζει τρία μοντέλα :

- Ένα ανεξάρτητο υπολογισμών μοντέλο (Computation Independent Model) το οποίο είναι η άποψη ενός συστήματος που δεν δείχνει λεπτομέρειες της δομής των συστημάτων και έχει ένα λεξικό οικείο καθώς χρησιμοποιείται για τον προσδιορισμό του συστήματος.
- Ένα μοντέλο ανεξάρτητο της πλατφόρμας (Platform Independent Model-PIM) το οποίο είναι η άποψη ενός συστήματος που από την μια πλευρά παρέχει ένα ιδιαίτερο τεχνικό προσδιορισμό του συστήματος , αλλά από την άλλη πλευρά επιδεικνύει ένα ιδιαίτερο βαθμό ανεξαρτησίας στην πλατφόρμα , τόσο όσο να γίνει κατάλληλο για χρήση με έναν διαφορετικό αριθμό πλατφορμών. Το σύστημα περιγράφεται σε μορφή ανεξάρτητο πλατφόρμας στο τέλος της φάσης σχεδίασης.
- Ένα μοντέλο ειδικής πλατφόρμας (Platform Specific Model-PSM) το οποίο είναι η άποψη ενός συστήματος που συνδυάζει τους προσδιορισμούς στο



PIM με τις λεπτομέρειες που εξειδικεύουν πώς αυτό το σύστημα χρησιμοποιεί μια ιδιαίτερου τύπου πλατφόρμα.

Η Οδηγούμενη από Μοντέλα Μηχανική βασίζεται βαθιά στον μετασχηματισμό μοντέλων [31].Ο μετασχηματισμός μοντέλων είναι η διαδικασία μετασχηματισμού ενός μοντέλου σε ένα άλλο.Οι απαιτήσεις για να επιτευχθεί αυτό είναι η ύπαρξη των μεταμοντέλων των μοντέλων και μια γλώσσα μετασχηματισμού στην οποία γράφονται οι κανόνες για μετασχηματισμό των στοιχείων του ενός μεταμοντέλου σε εκείνα του άλλου μεταμοντέλου.*Meta* είναι ένα πρόθεμα που προέρχεται από την Ελληνική λέξη «μετά» , και η οποία χρησιμοποιείται στην επιστημολογία και σημαίνει «σχετικά».

Στην περιοχή της μηχανικής λογισμικού ένα *μοντέλο* είναι μια αφαίρεση ενός συστήματος λογισμικού ή μέρους αυτού και ένα *μεταμοντέλο* είναι άλλη αφαίρεση , ορίζοντας τις ιδιότητες του μοντέλου μόνο του.Έτσι, όπως ένα πρόγραμμα υπολογιστή συμμορφώνεται με την γραμματική της γλώσσας προγραμματισμού στην οποία είναι γραμμένο , ένα μοντέλο συμμορφώνεται με το μεταμοντέλο του ή την αναφορά του μοντέλου του.Οπωσδήποτε , ακόμη και ένα μεταμοντέλο είναι από μόνο του ένα μοντέλο.Στην περιοχή της μηχανικής μοντέλων υπάρχει ακόμη ένα άλλο επίπεδο αφαίρεσης , το *μεταμεταμοντέλο* που ορίζεται ως το μοντέλο που συμμορφώνεται στον εαυτό του [17].Θα υιοθετήσουμε τους ακόλουθους τρεις ορισμούς από την ίδια εργασία:

Ορισμός 1. Ένα *μεταμεταμοντέλο* είναι ένα μοντέλο το οποίο είναι το μοντέλο της αναφοράς του (δηλ. συμμορφώνεται στον εαυτό του)

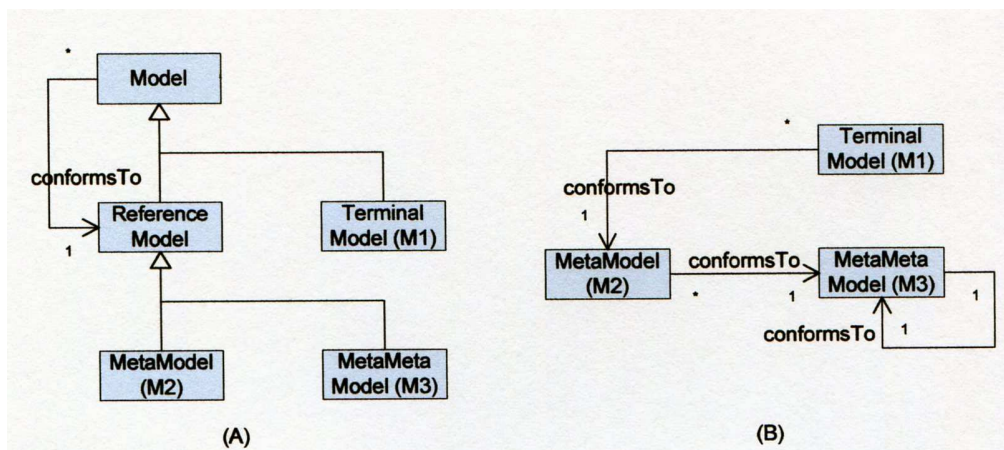
Ορισμός 2. Ένα *μεταμοντέλο* είναι ένα μοντέλο τέτοιο ώστε το μοντέλο αναφοράς του είναι ένα μεταμοντέλο.

Ορισμός 3. Ένα *τερματικό μοντέλο* είναι ένα μοντέλο τέτοιο ώστε το μοντέλο αναφοράς του είναι ένα μεταμοντέλο.

Καλούμε αυτά τα επίπεδα : M1, M2 και M3.Το M1 συνίσταται από όλα τα μοντέλα που δεν είναι μεταμοντέλα.Το M2 από όλα τα μεταμοντέλα που δεν είναι το μεταμεταμοντέλο.Το M3 από ένα μοναδικό μεταμεταμοντέλο για κάθε δεδομένο τεχνικό χώρο.Η εικ.1 (Α) δείχνει πώς να προσαρμόσουμε τον ορισμό του μοντέλου σε αυτόν τον τριών επιπέδων μοντελοποίησης σωρό.Η εικ.1 (Β) δείχνει τους συνδέσμους μεταξύ των



τριών επιπέδων μοντέλων σύμφωνα με τους παραπάνω ορισμούς.Σ' αυτή την εργασία η λέξη μοντέλο θα αναφέρεται συνήθως σε ένα τερματικό μοντέλο.



Εικ. 1. Παρουσίαση του σχήματος μεταμοντελοποίησης (A) με ορισμό μοντέλου (B).

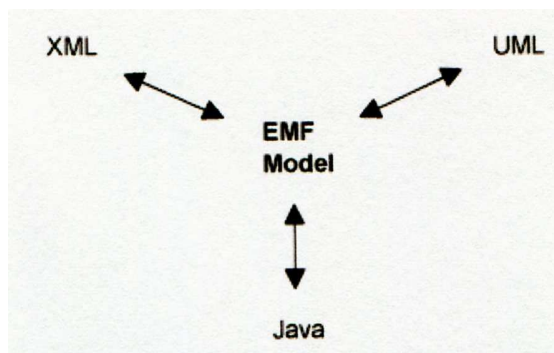
Η δομή για τα μοντέλα που ορίστηκε σ' αυτό το κεφάλαιο είναι συμβατό με την άποψη OMG όπως φαίνεται στον οδηγό της MDA [28]. Ένα παράδειγμα αυτής της προσέγγισης είναι ο τεχνικός χώρος της EBNF : προγράμματα (M1) προσκολλώνται σε γραμματικές (M2) , οι οποίες προσκολλώνται στην γραμματική του EBNF (M3).

2.1.1 Μετα- και μεταμετα – μοντέλα

Στην καρδιά της διαδικασίας του μετασχηματισμού μοντέλων βρίσκεται το Eclipse Modeling Framework (EMF) [5]. Το EMF ενοποιεί Java , XML και UML τεχνολογίες, επιτρέποντας στον σχεδιαστή να εναλλάσσεται μεταξύ τους καθώς παράγουν την ίδια πληροφορία σε μια διαφορετική παρουσίαση. Η εικ.2 δείχνει πώς το EMF ενοποιεί αυτές τις τρεις τεχνολογίες. Άσχετα από το ποιά χρησιμοποιείται για να το ορίσει , ένα EMF μοντέλο είναι η συνήθης υψηλού επιπέδου παρουσίαση που τις «κολλάει» όλες μαζί.

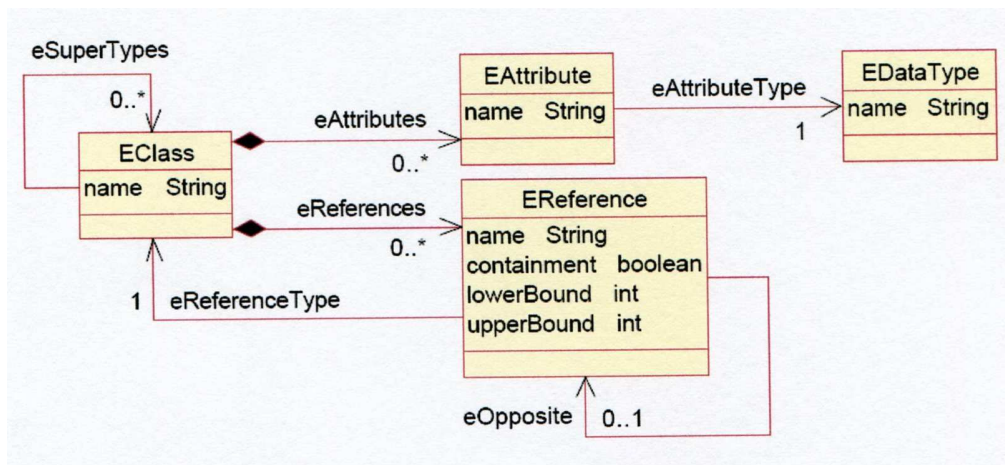


Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 2 . Το EMF μοντέλο ενοποιεί Java , XML και UML τεχνολογίες

Ecore είναι το EMF μοντέλο ενός μοντέλου (μεταμοντέλο).Χρησιμοποιώντας ecore, ένας σχεδιαστής μπορεί να ορίσει μοντέλα.Λειτουργεί ως ένα μεταμεταμοντέλο και χρησιμοποιείται για κατασκευή μεταμοντέλων.Ορίζει ότι ένα μοντέλο συντίθεται από στιγμιότυπα του τύπου EClass, τα οποία μπορούν να έχουν attributes (στιγμιότυπα του τύπου EAttribute) ή αναφορά σε στιγμιότυπα άλλης EClass (μέσω του τύπου EReference).Τελικά , τα EAttributes μπορούν να είναι ποικίλα EDataType στιγμιότυπα (όπως ακέραιοι , strings , πραγματικοί αριθμοί κλπ).Η εικ.3 δείχνει το ecore μεταμοντέλο λεπτομερώς.



Εικ. 3. Το ecore μεταμοντέλο

Μια όμοια τεχνολογία , η Meta-Object Facility (MOF) , είναι ένα Object Management Group (2001) πρότυπο για παρουσίαση και διαχείριση μεταμοντέλων.Υπάρχει ένας



αριθμός από ουσιώδη σχέδια στην MOF μοντελοποίηση. Ένα πακέτο χρησιμοποιείται για να υποκρύψει μια συλλογή από συσχετιζόμενες κλάσεις και συνδέσμους. Πακέτα μπορούν επίσης να περιέχουν ορισμούς απλού τύπου. Οι κλάσεις υπάρχουν με την κοινά χρησιμοποιούμενη έννοια της λέξης, περιγράφοντας ένα αντικείμενο και τις ιδιότητές του. Αυτές οι ιδιότητες παρουσιάζονται μέσω των Attributes και των References, οι οποίες μπορούν να ιεραρχηθούν χρησιμοποιώντας ένα πολλαπλής-ιεράρχησης σύστημα. Τα attributes έχουν ένα όνομα και έναν τύπο. Αυτό περιλαμβάνει μια σειρά τύπων από βασικούς τύπους όπως ακραίους, strings και Booleans σε πιο σύνθετους τύπους όπως οι δομημένοι τύποι. Επιπλέον, αμφότερα τα attributes έχουν κατώτερο και ανώτερο όριο στον αριθμό εμφάνισης με ένα στιγμιότυπο κλάσης. Ένας σύνδεσμος χρησιμοποιείται για να παρουσιάσει μια σχέση μεταξύ στιγμιότυπων των δύο κλάσεων, καθένα από τα οποία παίζει ένα ρόλο μέσα στον σύνδεσμο. Οι σύνδεσμοι μπορούν να έχουν την πρόσθετη ιδιότητα του «περιεχομένου» (containment) δηλ. ένας σύνδεσμος παρουσιάζει μια σχέση «περιεχομένου» αν μια από τις συμμετέχουσες κλάσεις δεν υπάρχει χωρίς την άλλη. Μια κλάση που συμμετέχει σε ένα σύνδεσμο μπορεί επίσης να περιέχει μια αναφορά (EReference) στον σύνδεσμο. Μια αναφορά εμφανίζεται πιο πολύ ως ένα attribute, αλλά αντανakλά το σύνολο των στιγμιότυπων κλάσης που συμμετέχει στον σύνδεσμο με το στιγμιότυπο της περιέχουσας κλάσης.

Η MOF είναι παλαιότερη από την EMF και αυτή επηρεάζει το σχέδιό της. Η MOF αρχικά σχεδιάστηκε πρωτίστως για χρήση με την αρχιτεκτονική CORBA (Common Object Request Broker Architecture) η οποία δίνει την δυνατότητα σε προγράμματα, που καλούνται objects, να επικοινωνήσουν με ένα άλλο άσχετα με την γλώσσα προγραμματισμού που είναι γραμμένα ή με το λειτουργικό σύστημα που αυτά τρέχουν.

Από την άλλη πλευρά, η EMF είναι ένα προϊόν του Eclipse project, ενός ανοικτού κώδικα project και προτάθηκε ως ένα χαμηλού κόστους εργαλείο που επιτυγχάνει τα οφέλη της τυπικής μοντελοποίησης και της παραγωγής java κώδικα. Συνεπώς, θα μπορούσε να πει κάποιος ότι η EMF κάνει μια από κάτω προς τα πάνω προσέγγιση ενώ η MOF από πάνω προς τα κάτω [12].

Οποσδήποτε, το EMF μεταμοντέλο είναι απλούστερο από εκείνο της MOF στις ιδιότητες και στην δομή περιεχομένου. Η τεχνολογία EMF σήμερα αποτελεί το de facto

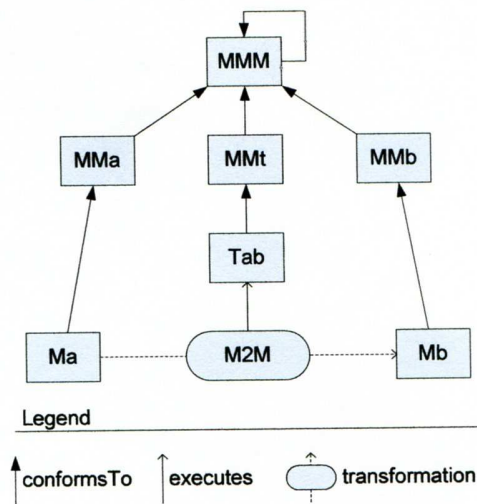


πρότυπο για την μεγάλη ανοικτού κώδικα κοινότητα και επί πλέον τρίτες ομάδες ορίζουν MDE εργαλεία βασισμένα στην EMF τεχνολογία , όπως πλατφόρμες για ανάπτυξη λογισμικού οδηγούμενη από μοντέλα (π.χ. openArchitectureWare-Oaw platform).Για όλους τους λόγους αυτούς αποφασίσθηκε ότι για την ASEME μεταμοντελοποίηση , θα χρησιμοποιηθεί η EMF τεχνολογία.

Παρακάτω παρουσιάζονται τα μεταμοντέλα της ASEME με λεπτομέρειες.Ακολουθώς περιγράφονται οι αυτόματοι μετασχηματισμοί μοντέλων παράλληλα με τις χρησιμοποιούμενες τεχνολογίες.

2.1.2 Μετασχηματισμοί μοντέλων

Μετά από τον ορισμό των μοντέλων των μοντέλων (ή μεταμοντέλων) είναι δυνατόν να ορίσουμε μετασχηματισμούς ενός μοντέλου σε κάποιο άλλο.Το γενικό σχήμα της διαδικασίας μετασχηματισμού που ακολουθείται τουλάχιστον από τις δύο κυριότερες γλώσσες μετασχηματισμού την ATLAS (Atlas Transformation Language) και την QVT (Query-View Transformation) παρουσιάζεται στην εικ. 4.Στην κορυφή υπάρχει ένα κοινό μεταμεταμοντέλο (MMM) στο οποίο συμμορφώνονται δύο μεταμοντέλα (MMa και MMB).Ο στόχος της διαδικασίας μετασχηματισμού ή μοντέλο σε μοντέλο διαδικασίας (M2M) είναι να πάρει ένα μοντέλο Ma , το οποίο συμμορφώνεται προς το MMa , ως είσοδο (ή μοντέλο πηγή) και παράγει το Mb , το οποίο συμμορφώνεται προς το MMB ως έξοδο (ή μοντέλο στόχος).Εκτός από τα μοντέλα πηγή και στόχος η διαδικασία εκτελεί ένα πρόγραμμα μετασχηματισμού (που θα καλείται Tab) και περιγράφει την διαδικασία μετασχηματισμού ενός μοντέλου που συμμορφώνεται προς το MMa σε ένα μοντέλο που συμμορφώνεται σε ένα μεταμοντέλο MMB.Το πρόγραμμα μετασχηματισμού είναι από μόνο του ένα μοντέλο που συμμορφώνεται σε ένα μεταμοντέλο (MMt) , το οποίο με την σειρά του συμμορφώνεται στο μεταμεταμοντέλο (MMM).Κατ' αυτόν τον τρόπο, όπως στην περίπτωση του EBNF, το MMt ορίζει την αφαιρετική σύνταξη της γλώσσας μετασχηματισμού.Αμφότερες οι ATL και QVT ορίζουν τις αφαιρετικές συντάξεις τους μέσω ενός τέτοιου μεταμοντέλου.



Εικ. 4 .Το γενικό σχήμα του μοντέλου μετασχηματισμού

Ακολούθως παρουσιάζεται μια αξιολόγηση και επέκταση των διαθέσιμων προσεγγίσεων, όσον αφορά κάποιες από τις γλώσσες μετασχηματισμού μοντέλων όπως εμφανίζεται στην βιβλιογραφία [7].

Οι μετασχηματισμοί μοντέλων είναι ένα κλειδί – προαπαιτούμενο για την Μηχανική Οδηγούμενη από Μοντέλα και γι' αυτό παρουσιάζει μια ενεργή ερευνητική περιοχή. Πολλές γλώσσες μετασχηματισμού μοντέλων είναι διαθέσιμες, ενώ οι γλώσσες μπορούν να κατηγοριοποιηθούν σε διαφορετικές προσεγγίσεις. Μια προσέγγιση μετασχηματισμού μοντέλων, ανάλογα από την εκάστοτε ιδιαίτερη κατάσταση, θα πρέπει να είναι κατά το καλύτερο δυνατόν κατάλληλη να εκπληρώσει την δεδομένη διαδικασία από κάποια άλλη προσέγγιση. Υπάρχουν ταξινομήσεις των γλωσσών μετασχηματισμού μοντέλων που περιλαμβάνουν μια αναφορά των γενικών χαρακτηριστικών της γλώσσας που πρέπει να υποστηρίξουν. Είναι δυνατόν να συγκρίνουμε προσεγγίσεις μετασχηματισμού μοντέλων με σκοπό να βρούμε πόσο είναι διαθέσιμες για ένα δεδομένο πρόβλημα. Όπως στις κοινές προσεγγίσεις του αντικειμενοστραφούς προγραμματισμού, όπως στην java, υπάρχουν μερικά ιδιαίτερα προβλήματα που εμφανίζονται επαναληπτικά. Για παράδειγμα, είναι συχνά αναγκαίο να μετασχηματίσουμε μια τιμή ενός attribute στο μοντέλο πηγή σε ένα αντικείμενο στο μοντέλο στόχος. Για τέτοιες περιπτώσεις, θα μπορούσαμε να θεωρήσουμε ότι τέτοια



προβλήματα λύνονται με έναν γενικό τρόπο.Επιπλέον , γενικές λύσεις μπορούν να επαναχρησιμοποιηθούν σε άλλους μετασχηματισμούς μοντέλων.Στην αναφορά αυτή εκτιμήθηκαν τέσσερις γλώσσες μετασχηματισμού μοντέλων με ονόματα ATL , SmartQVT , Kermeta και Triple Graph Grammas (TGG)(χρησιμοποιώντας το MOFLON ως εργαλείο μετασχηματισμού).Αυτές οι γλώσσες επιλέχθηκαν γιατί παρουσιάζουν την κατάσταση της εργασίας στον μετασχηματισμό μοντέλων.Οι ATL και SmartQVT είναι υβριδικές (ένα μίγμα δηλωτικής και «επιτακτικής») γλώσσες.Επιπλέον ιδιαίτερα ,η SmartQVT υλοποιεί το «λειτουργικό» κομμάτι του προτύπου OMG QVT.Η Kermeta δρά ως παράδειγμα για μια «επιτακτική» γλώσσα , ενώ οι TTGs παρουσιάζουν μια δηλωτική προσέγγιση με μια γραφική σύνταξη.Για τον σκοπό της εκτίμησης ορίστηκαν και εκτιμήθηκαν χρησιμοποιώντας τις παραπάνω γλώσσες , διάφορα παραδείγματα μετασχηματισμού μοντέλων.Τα παραδείγματα αυτά καλύπτουν προβλήματα μετασχηματισμού μοντέλων που εμφανίζονται στην πράξη , προκειμένου να δώσουν έμφαση στα πλεονεκτήματα και τους περιορισμούς της συγκεκριμένης γλώσσας.Επίσης χρησιμοποιήθηκε η Kermeta για να υλοποιήσει μια βιβλιοθήκη που επιλύει κοινά προβλήματα που εμφανίζονται στους μετασχηματισμούς αυτούς.Η βιβλιοθήκη αυτή μπορεί μετά να χρησιμοποιηθεί σε ειδικούς μετασχηματισμούς για απλοποίηση της διαδικασίας ανάπτυξης.Τελικά τα αποτελέσματα της εκτίμησης διερμηνεύθηκαν και συζητήθηκαν με σκοπό να αποτελέσουν ένα οδηγό για το ποιές προσεγγίσεις μετασχηματισμού μοντέλων είναι κατάλληλες για ποιο πρόβλημα.Ακολούθως παρατίθεται μια περιεκτική αναφορά στην εκτίμηση αυτή :

ATL

Επισκόπηση

- Αναπτύχθηκε από την ATLAS INRIA & LINA
- Δηλωτική προσέγγιση με «προστακτικά» πρόσθετα (υβριδική προσέγγιση)
- Βασισμένο στο Eclipse IDE

Πλεονεκτήματα

- Εύκολη στην εκμάθηση
- Καλή IDE ολοκλήρωση
- Καλά αναγνώσιμος κώδικας

Μειονεκτήματα



Αργυρίου Αντώνιος, ΈΜετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

- Κάποια προβλήματα μετασχηματισμού δεν λύνονται προαισθητικά με δηλωτική προσέγγιση
- Τα μοντέλα εξόδου δεν ελέγχθηκαν για συμμόρφωση προς τα μεταμοντέλα

SmartQVT

Επισκόπηση

- Αναπτύχθηκε από την Franch Telecom R&D
- «Επιτακτική» προσέγγιση
- Υλοποίηση του «σχεσιακού» QVT προτύπου
- Βασισμένο στο Eclipse IDE

Πλεονεκτήματα

- Συμμόρφωση προς QVT
- Ισχυρά MT χαρακτηριστικά
- Καλή IDE ολοκλήρωση
- Επεκτάσιμη αρχιτεκτονική

Μειονεκτήματα

- Δυσκολότερη στη εκμάθηση από άλλες γλώσσες

Kermeta

Επισκόπηση

- Αναπτύχθηκε από την Triskell/IRISA
- «Επιτακτική» προσέγγιση
- Γενική γλώσσα προγραμματισμού
- Βασισμένο στο Eclipse IDE

Πλεονεκτήματα

- Ισχυρά χαρακτηριστικά γλώσσας
- Καλή IDE ολοκλήρωση

Μειονεκτήματα

- Μη υποστήριξη built-in για βασικά προβλήματα μετασχηματισμού (tracing)
- Απαιτείται περισσότερος κώδικας μετασχηματισμού σε σχέση με άλλες γλώσσες

ModelMorf

Επισκόπηση

- Αναπτύχθηκε από την Tata Research-Κέντρο Ανάπτυξης και Σχεδίασης
- Υλοποίηση του «σχεσιακού» QVT προτύπου



- Φτωχή «δηλωτικά»
- Standalone εφαρμογή

Πλεονεκτήματα

- Συμμόρφωση προς QVT
- Πολυκατευθυνόμενη
- Αυξητικότητα στόχων

Μειονεκτήματα

- Κάποια προβλήματα μετασχηματισμού είναι δύσκολο να λυθούν με «δηλωτικές» προσεγγίσεις
- Απουσία IDE ολοκλήρωσης
- Οι μετασχηματισμοί είναι δύσκολο να εκσφαλματωθούν

Αποτελέσματα εκτίμησης

Εκβάσεις

- Καμία «τέλεια» γλώσσα προγραμματισμού
- Η πιο κατάλληλη γλώσσα εξαρτάται από την περίπτωση
- Ποια χαρακτηριστικά θα χρειασθούμε;
- Εύρεση αποτελεσμάτων εκτίμησης για σύγκριση χαρακτηριστικών γλωσσών :
(<http://www.adore.at/mtcomp/>)

Γενικά ευρήματα

- Οι «δηλωτικές» προσεγγίσεις είναι καταλληλότερες για λιγότερο σύνθετα προβλήματα όπου απαιτείται πολυκατευθυνσιμότητα
- Οι «επιτακτικές» προσεγγίσεις είναι καταλληλότερες αν ο μετασχηματισμός είναι πιο σύνθετος
- Όλες οι γλώσσες μετασχηματισμού έχουν περιορισμούς στις διαθέσιμες βιβλιοθήκες και στους τελεστές υψηλότερου επιπέδου μετασχηματισμού

Επέκταση : Βιβλιοθήκη Kermeta για επαναχρησιμοποιούμενους μετασχηματισμούς

Προβλήματα

- Συχνά συμβαίνουν αντίγραφα μετασχηματισμού
- Συσσωμάτωση , κατάρρευση ιεραρχίας, Ανάστροφη ιδιοκτησία,....

Προσέγγιση

- Τα προβλήματα λύνονται με γενικό τρόπο
- Κατασκευάστηκε και διατίθεται ως βιβλιοθήκη Kermeta



Αποτελέσματα

- Απαιτείται πάνω από 20% λιγότερος κώδικας
- Αλλά ο κώδικας είναι δυσκολότερο να διαβασθεί

Άποψη

- Βελτίωση της αναγνωσιμότητας με υλοποίηση μιας βασισμένης σε API εύχερης διασύνδεσης
- DSL γλώσσα (π.χ. ATLAS,...) για επαναχρησιμοποίησιμα υποδείγματα.

Για τον μετασχηματισμό μοντέλο σε μοντέλο (M2M) χρησιμοποιήθηκε η γλώσσα μετασχηματισμού Atlas (ATL).Μια άλλη εναλλακτική , ως προς την Atlas επιλογή θα μπορούσε να ήταν η γλώσσα QVT.Η Atlas είναι καλύτερα αναπτυγμένη στο διαδίκτυο με οδηγό χρήσης και παραδείγματα , ενώ η μόνη πηγή για την QVT ήταν η παρουσίασή της.Γι' αυτό και καθώς οι απαιτήσεις αμφοτέρων των γλωσσών (ATL και QVT) είναι οι ίδιες [18,19] , η απόφαση ήταν να επιλεγεί η καλύτερα αναπτυγμένη στο διαδίκτυο.Ακολουθως αναφέρουμε διεξοδικότερα τα χαρακτηριστικά της ATL :

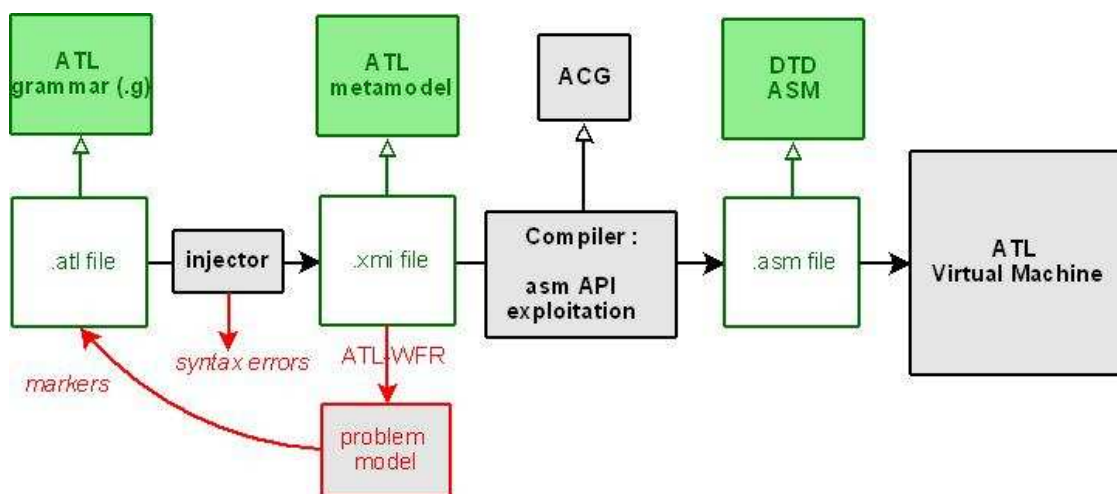
- Αναπτύχθηκε ως ένα τμήμα της πλατφόρμας AMMA (ATLAS Model Management Architecture)
- Είναι μια γλώσσα μετασχηματισμού σε μορφή κειμένου με λειτουργική σημασιολογία εκτέλεσης πηγής-στόχου
- Τα μοντέλα-πηγή είναι μόνο για ανάγνωση και δεν μπορούν να τροποποιηθούν , ενώ τα μοντέλα-στόχος είναι μόνο εγγράψιμα
- Η γλώσσα έχει δηλωτικά και επιτακτικά χαρακτηριστικά (υβριδική)
- Συνιστάται η δηλωτική μορφή της , τμήμα της οποίας είναι οι κανόνες αντιστοίχισης (matched rules) με αυτόματη αντιστοίχιση των παραγόμενων στοιχείων και η χωρίς παρενέργειες πλοήγηση με OCL (ObjectConstraintLanguage) 2.0
- Τα επιτακτικά της τμήματα είναι οι κανόνες επίκλησης (called rules) και τα block δράσης
- Ένας κανόνας αντιστοίχισης σχηματοποιείται στο σχέδιο-πηγή που αντιστοιχεί στο μοντέλο-πηγή και στο σχέδιο-στόχος που αντιστοιχεί στο μοντέλο-στόχος για κάθε αντιστοίχιση κατά την διάρκεια της εφαρμογής του κανόνα



- Κατά την εκτέλεση των μετασχηματισμών με τους κανόνες αντιστοίχισης
 - ο Βρίσκουμε όλες τις πιθανές αντιστοιχίσεις στο μοντέλο-πηγή
 - ο Δημιουργούμε σχέδια-στόχου στο μοντέλο-στόχος
 - ο Αρχικοποιούμε ιδιότητες και συνδέσμους στο μοντέλο-στόχος
- Η σειρά εκτέλεσης των κανόνων αντιστοίχισης είναι μη αιτιοκρατική
- Το αποτέλεσμα είναι ακόμα αιτιοκρατικό εξαιτίας της σημασιολογίας εκτέλεσης των μοντέλων πηγής-στόχου (μοντέλα πηγής μόνο αναγνώσιμα και μοντέλα στόχου μόνο εγγράψιμα)
- Εκτός από τους κανόνες , η ATL επιτρέπει να ορισθούν και οι λεγόμενοι αρωγοί (helpers) , λειτουργικοί που χρησιμοποιούνται για να διατρέξουν όλο το μοντέλο-πηγή και οι των ιδιοτήτων που χρησιμοποιούνται για να συνδέσουν τις παραγόμενες μόνο για ανάγνωση τιμές , στα στοιχεία του μοντέλου πηγής

Η ATL υποστηρίζεται από ένα σύνολο εργαλείων ανάπτυξης όπως είναι ένας συντάκτης, ένας μεταγλωττιστής , μια εικονική μηχανή και έναν διορθωτή. Οι ATL μετασχηματισμοί μεταγλωττίζονται σε προγράμματα εξειδικευμένα σε byte-κώδικα που εκτελείται από την ATL εικονική μηχανή , η οποία εξειδικεύεται στον χειρισμό μοντέλων και παρέχει ένα σύνολο από οδηγίες για τον χειρισμό τους.

Η εικ. 5 παρουσιάζει την διαδικασία της ATL μεταγλώττισης δηλ. περιγράφει τα συστατικά και τον ρόλο τους στην εκτέλεση του μετασχηματισμού.



Εικ. 5. Η πορεία της εκτέλεσης του ATL μετασχηματισμού



2.2 Μεθοδολογία Δημιουργίας Πρακτοροστραφών Συστημάτων (ASEME)

Η μεθοδολογία δημιουργίας πρακτοροστραφών συστημάτων είναι μια διαδικασία ανάπτυξης λογισμικού οδηγούμενη από μοντέλα η οποία χρησιμοποιεί την γλώσσα μοντελοποίησης πρακτόρων (Agent Modeling Language-AMOLA) , για μοντελοποίηση συστημάτων πολυ-πρακτόρων. Η AMOLA παρέχει το συντακτικό και την σημειολογία για την δημιουργία μοντέλων συστημάτων πολυ-πρακτόρων καλύπτοντας τις φάσεις της ανάλυσης και της σχεδίασης της διαδικασίας ανάπτυξης λογισμικού. Υποστηρίζει μια προσέγγιση σχεδιασμού πρακτόρων με modules και εισάγει τις έννοιες του ενδο-πρακτορικού (Intra-AgentControl) και δια-πρακτορικού (interAgentControl) ελέγχου. Ο πρώτος ορίζει την συμπεριφορά του πράκτορα συντονίζοντας τα διαφορετικά modules που υλοποιούν τις δραστηριότητές του , ενώ η δεύτερη ορίζει τα πρωτόκολλα που κυβερνούν τον συντονισμό της κοινωνίας των πρακτόρων. Η φάση της ανάλυσης χτίζεται πάνω στις έννοιες της δραστηριότητας και της λειτουργικότητας. Η AMOLA συμφωνεί και με την ατομική και με την κοινωνική άποψη των πρακτόρων , δείχνοντας πώς τα πρωτόκολλα και οι δραστηριότητες μπορούν να ολοκληρωθούν σ' ένα σχέδιο πρακτόρων. Αυτή είναι η πρώτη πρωτοτυπία της ASEME , το γεγονός δηλ. ότι τα μοντέλα του δια-πρακτορικού και του ενδο-πρακτορικού ελέγχου ορίζονται χρησιμοποιώντας τον ίδιο φορμαλισμό επιτρέποντας έτσι την ολοκλήρωση των δια-πρακτορικών πρωτοκόλλων στο μοντέλο του πράκτορα , ως δραστηριότητες. Η διαδικασία ανάπτυξης περιγράφεται χρησιμοποιώντας το μεταμοντέλο Software Process Engineering Metamodel (SPEM). Το ανεξάρτητο από πλατφόρμα μοντέλο (Platform Independent Model-PIM) της το οποίο είναι η έξοδος της φάσης σχεδίασης , είναι ένα διάγραμμα καταστάσεων που μπορεί να στιγμιτυποποιηθεί σ' ένα αριθμό πλατφορμών χρησιμοποιώντας υπάρχοντα CASE εργαλεία και σε μια πλατφόρμα πρακτόρων , το πλαίσιο ανάπτυξης java πρακτόρων (Java Agent Development Framework -JADE). Η διαδικασία της ASEME είναι επαναληπτική , επιτρέποντας αυξανόμενη ανάπτυξη και παρέχει την πρωτότυπη δυνατότητα να μετακινηθεί προς τα πίσω σε κάθε προηγούμενη

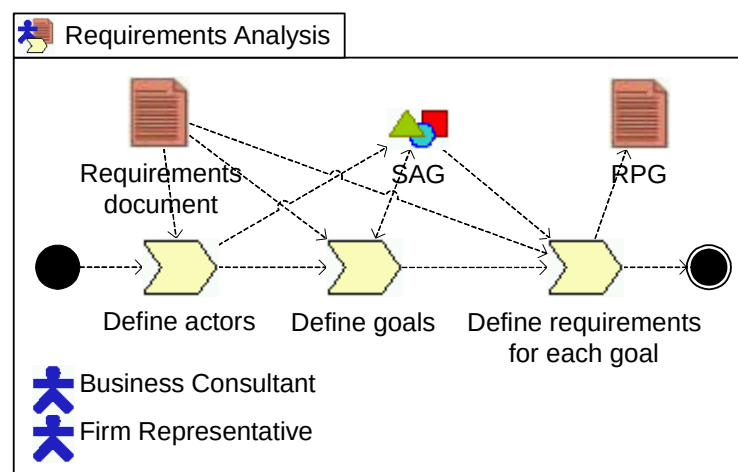


Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

φάση λόγω της χρησιμοποιούμενης MDE προσέγγισης. Υπάρχουν έξι φάσεις ανάπτυξης , οι τέσσερις πρώτες παράγουν μοντέλα συστήματος (φάσεις ανάπτυξης) , ενώ οι άλλες δύο (αξιολόγηση και βελτιστοποίηση) αξιολογούν και βελτιστοποιούν αυτά τα μοντέλα. Ακολουθώς παρουσιάζονται μια επισκόπηση της ASEME, καθώς και οι φάσεις της ανάλυσης απαιτήσεων, της ανάλυσης και της σχεδίασης συστήματος (βλ. εικ. 6, 7, 8, 9).

Φάση Ανάπτυξης	Επίπεδα Αφαίρεσης		
	Επίπεδο Κοινωνίας Πρακτόρων	Επίπεδο Πράκτορα	Επίπεδο Ικανότητας
Ανάλυση απαιτήσεων Μοντέλα AMOLA	Δράστες Διάγραμμα Δραστήων-Στόχων (SAG)	Στόχοι Διάγραμμα Δραστήων-Στόχων (SAG)	Απαιτήσεις Απαιτήσεις Στόχων
Ανάλυση Συστήματος Μοντέλα AMOLA	Ρόλοι και Πρωτόκολλα Διάγραμμα Περίπτωσης Χρήσης (SUC), Πρωτόκολλα Άλληλεπίδρασης Πρακτόρων (AIP)	Ικανότητες Διάγραμμα Περίπτωσης Χρήσης (SUC), Μοντέλο Ρόλων (SRM)	Λειτουργίες Πίνακας Λειτουργιών
Σχεδίαση Συστήματος Μοντέλα AMOLA	Λειτουργία Κοινωνίας Πρακτόρων Δια-πρακτορικός έλεγχος (EAC), Οντολογία, Τύποι Μηνυμάτων	Λειτουργία Πράκτορα Ενδο-πρακτορικός έλεγχος (IAC)	Συστατικά μέρη
Υλοποίηση	Κώδικας Διαχείρισης Πλατφόρμας	Κώδικας πράκτορα	Υλοποίηση ικανοτήτων
Επαλήθευση συστήματος	Έλεγχος λειτουργίας πρωτοκόλλων	Έλεγχος λειτουργίας πράκτορα	Έλεγχος συστατικών μερών
Βελτιστοποίηση	Αριθμός στιγμιότυπων πρακτόρων	Πόροι πρακτόρων	Βελτιστοποίηση αλγορίθμων

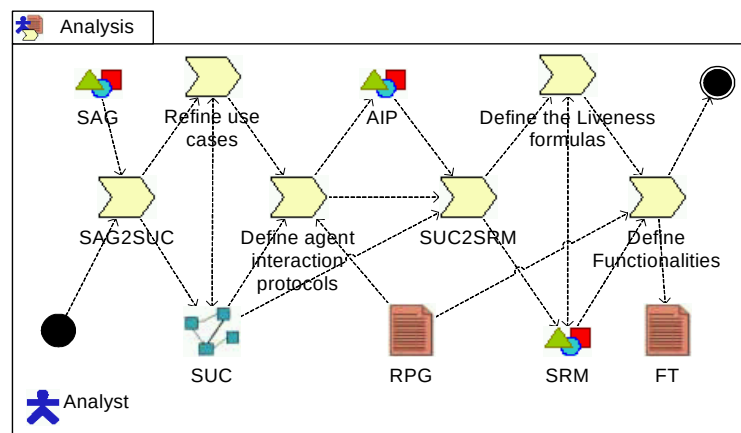
Εικ. 6. Οι φάσεις της μεθοδολογίας ASEME και τα AMOLA προϊόντα τους



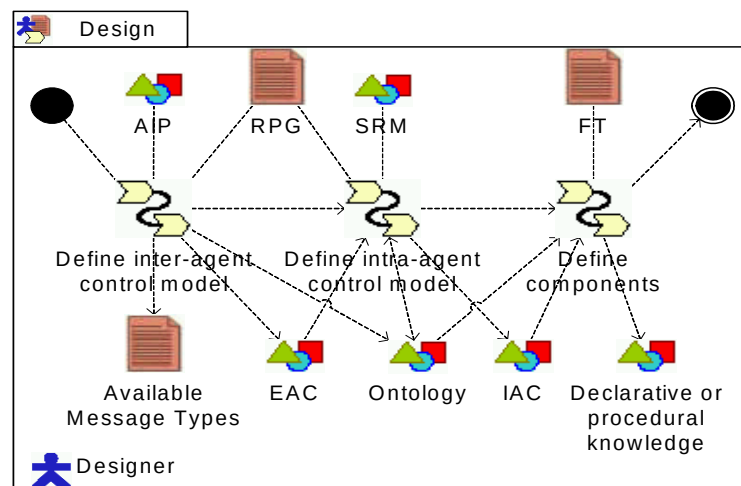
Εικ. 7. Η φάση ανάλυσης απαιτήσεων της ASEME



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 8. Η φάση ανάλυσης συστήματος της ASEME



Εικ. 9. Η φάση σχεδίασης συστήματος της ASEME

Η παράγραφος αυτή έχει σαν σκοπό να δείξει τις τεχνικές λεπτομέρειες υλοποίησης των μετασχηματισμών μοντέλων και ότι συμβαίνει στην ASEME διαδικασία. Οι μετασχηματισμοί μοντέλων επιτρέπουν σε μοντέλα προηγούμενης φάσης να μετασχηματιστούν σε μοντέλα μιας επόμενης φάσης, γεγονός που αποτελεί την δεύτερη πρωτοτυπία της ASEME. Η απαίτηση για τον ορισμό μιας διαδικασίας μετασχηματισμού μοντέλων είναι η ύπαρξη των μεταμοντέλων που περιγράφουν τα μοντέλα πηγή και στόχου.

Υπάρχουν τέσσερις τύποι τεχνικών μετασχηματισμού [21] καθένας από τους οποίους σχετίζεται με διαφορετικές τεχνολογίες:

- Μοντέλο σε μοντέλο (Model to Model-M2M) μετασχηματισμός. Αυτό το είδος μετασχηματισμού χρησιμοποιείται για μετασχηματισμό ενός τύπου γραφικού



- μοντέλου σε έναν άλλο τύπο γραφικού μοντέλου. Ένα παράδειγμα τέτοιου μετασχηματισμού είναι ο SAG2SUC μετασχηματισμός , όπου ένα SAG (System Actors Goals) διάγραμμα μετασχηματίζεται σε ένα SUC (System Use Case) διάγραμμα. Ο M2M μετασχηματισμός βασίζεται στα μεταμοντέλα πηγής και στόχου και ορίζει τους μετασχηματισμούς στοιχείων του μοντέλου πηγή σε στοιχεία του μοντέλου στόχου.
- Κείμενο σε μοντέλο (Text to Model-T2M) μετασχηματισμός. Αυτό το είδος μετασχηματισμού χρησιμοποιείται για μετασχηματισμό μιας παρουσίασης κειμένου σε γραφικό μοντέλο. Η παρουσίαση κειμένου πρέπει να βασίζεται σε ένα ορισμένο συντακτικό γλώσσας που συνήθως χρησιμοποιεί BNF ή EBNF (Extended Backus- Naur Form) , μια μετασύνταξη ή ένα μεταμεταμοντέλο γραμματικής. Η liveness formula προτείνει ένα τέτοιο είδος σύνταξης. Το γραφικό μοντέλο πρέπει να έχει ένα μεταμοντέλο. Τότε , ένας μετασχηματισμός του κειμένου σε ένα γραφικό μοντέλο μπορεί να ορισθεί , όπως στην περίπτωση του SRM2IAC μετασχηματισμού.
 - Μοντέλο σε Κείμενο (Model to Text-M2T) μετασχηματισμός. Τέτοιοι μετασχηματισμοί χρησιμοποιούνται για μετασχηματισμό μιας οπτικής παρουσίασης σε κώδικα (κώδικας είναι κείμενο). Πάλι , η σύνταξη της γλώσσας-στόχου πρέπει να ορισθεί παράλληλα με το μεταμοντέλο του γραφικού μοντέλου.
 - Κείμενο σε Κείμενο (Text to Text-T2T) μετασχηματισμός. Τέτοιοι μετασχηματισμοί χρησιμοποιούνται για μετασχηματισμό μιας παρουσίασης κειμένου σε μια άλλη παρουσίαση κειμένου. Αυτή είναι συνήθως η περίπτωση , όπου ένα πρόγραμμα γραμμένο για μια ειδική γλώσσα προγραμματισμού μετασχηματίζεται σε ένα πρόγραμμα μιας άλλης γλώσσας προγραμματισμού.

Η παρούσα εργασία θα ασχοληθεί μόνο με το πρώτο τύπο μετασχηματισμών , τον τύπο M2M.

2.2.1 Μεταμοντέλα

Οι Bernon et. al. (2005) [2] παρουσίασαν τα μεταμοντέλα των πιο δημοφιλών μεθοδολογιών AOSE (Agent Oriented Software Engineering-Πρακτοροστραφής Μηχανική Λογισμικού). Αυτά περιλαμβάνουν μεταμοντέλα για τις μεθοδολογίες Gaia και



Tropos.Αυτά τα μοντέλα χρησιμοποιήθηκαν σαν μια βάση για τον ορισμό των μεταμοντέλων SAG και SRM.Τα μεταμοντέλα SUC και IAC ήταν κύρια εμπνευσμένα από το UML μεταμοντέλο.Τα υπάρχοντα μεταμοντέλα μπορούν να βρεθούν σε μορφή.ecore (και σε έναν αριθμό από άλλα μορφή επίσης) στην zoos αποθήκη της AtlanMod , που βρίσκεται στην πόλη Νάντ της Γαλλίας.

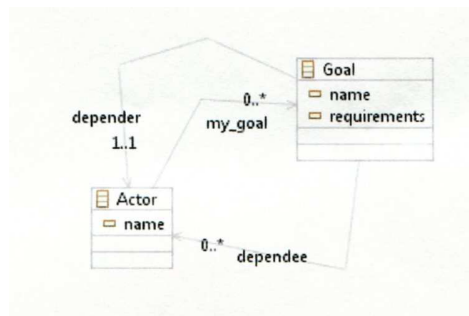
Οι επόμενες παράγραφοι ορίζουν τα μεταμοντέλα της AMOLA (Agent Modeling Language-Γλώσσα Μοντελοποίησης Πρακτόρων).Κάθε φορά παρουσιάζεται και το μεταμοντέλο που ενέπνευσε το AMOLA μεταμοντέλο.Αυτό γίνεται γιατί αφ' ενός ο αναγνώστης μπορεί να δει και μόνος του ότι τα μοντέλα ακολουθούν το πρωτότυπο μεταμοντέλο μπορούν να μετασχηματιστούν σε AMOLA μεταμοντέλα με μια απευθείας διαδικασία (που επιτυγχάνεται ακριβώς με ευθυγράμμιση σχεδίων) και έτσι με την μέθοδο αυτή τμήματα μπορούν εύκολα να εισαχθούν στην AMOLA διαδικασία και αφ' ετέρου , η απλότητα των μοντέλων της AMOLA θα γίνεται καταφανής , δείχνοντας ότι αυτό είναι ιδανικό για ευκίνητη ανάπτυξη.Τα μεταμοντέλα παρουσιάζονται σε εικόνες έτσι ώστε να γίνουν καλύτερα κατανοητά από τον αναγνώστη.Αυτά περιλαμβάνονται οπωσδήποτε και στο.ecore XML μορφή , στην τροποποιημένη τους μορφή για τις ανάγκες της εργασίας στο παράρτημα Δ.Τα γραφικά μοντέλα δημιουργούνται αυτόματα χρησιμοποιώντας τα Eclipse EMF εργαλεία.

Μοντέλο Συστήματος Δραστών-Στόχων- SAG (System Actors Goals Model)

Το SAG μοντέλο είναι ένα υποσύνολο του μοντέλου Δραστών του Tropos.ecore μοντέλου.Είναι το ίδιο μεταμοντέλο που εμφανίζεται και στο [36]. Καθώς το AMOLA SAG διάγραμμα δεν χρησιμοποιεί όλα αυτά τα σχέδια μία πιο συμπαγής έκδοση του σχεδίου μεταμοντέλου δραστών του Tropos έχει ορισθεί και παρουσιάζεται στην εικ. 10. Έτσι υπάρχουν τα σχέδια Δραστών και Στόχων.Ο δράστης αναφέρεται στους στόχους του χρησιμοποιώντας το *EReference my_goal*, ενώ ο Στόχος αναφέρει ένα μοναδικό depender (εξαρτώμενος) και κανέναν ή περισσότερους dependees(εξαρτητές). Ο αναγνώστης θα πρέπει να σημειώσει την επιλογή να προσθέσουμε το *EAttribute requirements* του στόχου όπου η πληροφορία απαιτήσεις ανά στόχο 'Requirements Per Goal' (RPG) αποθηκεύεται.



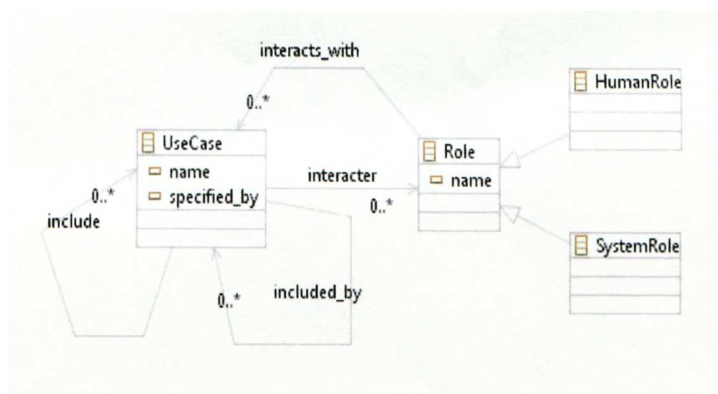
Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 10. Το AMOLA SAG μεταμοντέλο

Μοντέλο Συστήματος Περιπτώσεων Χρήσης – SUC (System Use Cases)

Η ίδια διαδικασία ακολουθήθηκε για τον ορισμό του μεταμοντέλου SUC. Ένα συμπαγές μεταμοντέλο που περιέχει τα σχέδια που χρησιμοποιούνται από την AMOLA έχει ορισθεί χρησιμοποιώντας το.ecore μεταμοντέλο και παρουσιάζεται στην εικ.11. Αυτό ακολουθεί τους ορισμούς χρησιμοποιώντας ελεύθερο κείμενο. Το σχέδιο *UseCase* (περίπτωση χρήσης) έχει ορισθεί ότι μπορεί να περιλαμβάνει και να περιλαμβάνεται από άλλα σχέδια *UseCase*. Αλληλεπιδρά με έναν ή περισσότερους ρόλους, που μπορεί να είναι Human ρόλοι (HumanRole) ή ρόλοι πρακτόρων (SystemRole).



Εικ. 11. Το AMOLA SUC μεταμοντέλο

Μοντέλο Συστήματος Ρόλων -SRM (System Roles Model)

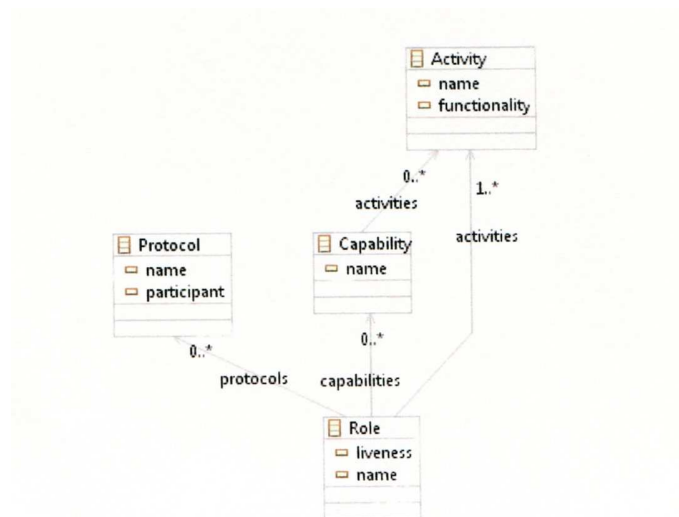
Το SRM μοντέλο είναι ένα υποσύνολο του GAIA 1.0.ecore μοντέλου όπως αυτό βρίσκεται αρχειοθετημένο στην zoos αποθήκη του AtlanMod(2008). Ένα πιο συμπαγές μεταμοντέλο που περιέχει τα σχέδια που χρησιμοποιούνται από την AMOLA έχει



ορισθεί χρησιμοποιώντας το *ecore* μεταμοντέλο και παρουσιάζεται στην εικ. 12. Το SRM μεταμοντέλο ορίζει το σχέδιο *Role* που αναφέρεται τα σχέδια :

- *Activity*(δραστηριότητα), που αναφέρεται σε μια απλή δραστηριότητα με 2 attributes (ιδιότητες) , το όνομα της (*name*) και την λειτουργικότητα της (functionality) δηλ. την περιγραφή του τί κάνει αυτή η δραστηριότητα.
- *Capability* (ικανότητα) , που αναφέρεται στις ομάδες των δραστηριοτήτων (στις οποίες αναφέρεται) πετυχαίνοντας έναν υψηλού επιπέδου στόχο, και
- *Protocol* (πρωτόκολλο) , με τις ιδιότητες του *name* και *participant* να αναφέρονται στα σχετικά τμήματα του μοντέλου EAC (μοντέλο διαπρακτορικού ελέγχου).

Το σχέδιο *Role* επίσης έχει ως ιδιότητες τα *name* (ως πρώτο) και *liveness* (*liveness formula* ως δεύτερο). Ο αναγνώστης θα μπορέσει να σημειώσει ότι η ιδιότητα (attribute) *functionality* του σχεδίου *Activity* ενσωματώνει την πληροφορία που περιλαμβάνεται στον AMOLA Functionality Table (Πίνακας λειτουργικότητας) , ο οποίος συνδέει κάθε δραστηριότητα με μια μοναδική λειτουργικότητα



Εικ. 12. Το AMOLA SRM μεταμοντέλο

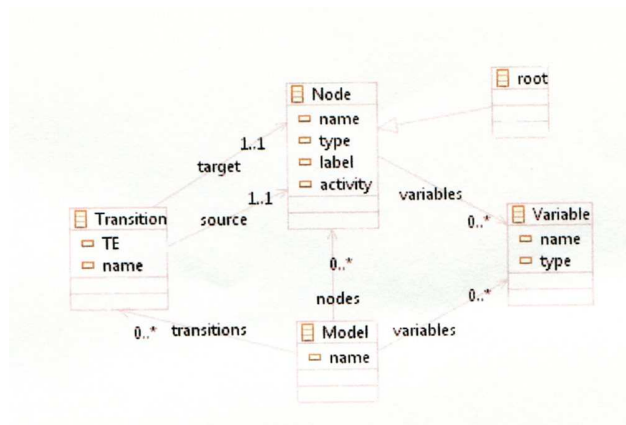
Μοντέλο Ενδοπρακτορικού Ελέγχου – IAC (Intra-Agent Control Model)

Η έμπνευση για το ορισμό του IAC μεταμοντέλου προέρχεται κύρια από τον ορισμό του UML διαγράμματος καταστάσεων. Σκοπεύοντας να ορίσουμε το διάγραμμα



καταστάσεων χρησιμοποιώντας τον AMOLA ορισμό του, το IAC μεταμοντέλο διαφέρει σημαντικά από το UML διάγραμμα καταστάσεων (βλ. εικ. 13). Οποσδήποτε ένα UML διάγραμμα καταστάσεων μπορεί να μετασχηματισθεί σε ένα IAC διάγραμμα καταστάσεων, αν και κάποια στοιχεία θα είναι δύσκολο να ορισθούν (η UML δεν υποστηρίζει εκφράσεις μετάβασης (transitions) και συνδέσμους (associations) των μεταβλητών στους κόμβους (nodes)).

Έτσι, το IAC μεταμοντέλο ορίζει ένα σχέδιο *Model* με EAttribute *name*- το οποίο χρησιμοποιείται για να ορίσει το namespace του IAC μοντέλου που θα πρέπει να ακολουθεί το java ή C# namespace φορμά των μοντέρνων πακέτων- και περιέχει nodes, transitions και EReference *variables*.



Εικ. 13. Το AMOLA IAC μεταμοντέλο

Τα nodes περιέχουν τα ακόλουθα attributes :

- *name* (Name). Το όνομα του node (κόμβου),
- *type*(λ). Ο τύπος του node (ένας απ' τους AND, OR, BASIC, START, END)
- *label*(label). Η ετικέτα του node, και
- *activity*(Activity). Η δραστηριότητα που αναφέρεται στο node

Τα nodes αναφέρονται σε *variables*. Η variable EClass έχει τα attributes *name* και *type*. Τελικά τα transitions έχουν 4 attributes:

- *name*, συνήθως με την μορφή <source node label >TO<target node label>



- *TE* , η έκφραση της transition που ακολουθεί την σχετική γραμματική
- *Source* ,ο node πηγή , και
- *Target* ,ο node στόχος

2.2.2 Μετασχηματισμοί μοντέλων

Τρεις τύποι τεχνικών μετασχηματισμού χρησιμοποιούνται στην ASEME διαδικασία. Κατά πρώτον , συμβαίνουν δύο M2M μετασχηματισμοί , SAG2SUC (διαβάζεται SAG to SUC) και SUC2SRM, κατόπιν ένας T2M , ο SRM2IAC και τελικά ένας M2T , ο IAC2JADE. Κάθε τύπος μετασχηματισμού απαιτεί χρήση διαφορετικής τεχνικής , περιλαμβανομένων γλώσσας προγραμματισμού , χρήση εργαλείων και εμπειρία στην δημιουργία σχετικών projects. Οποσδήποτε , τώρα που τα εργαλεία έχουν δημιουργηθεί στο eclipse αυτός που αναπτύσσει την ASEME μπορεί να επιτύχει τους μετασχηματισμούς μόνο με ένα «κλικ». Σ' αυτό το σημείο θα περιγράψουμε πώς τα εργαλεία μετασχηματισμού εφαρμόζονται , για τις ανάγκες της εργασίας αυτής.

M2M Μετασχηματισμοί

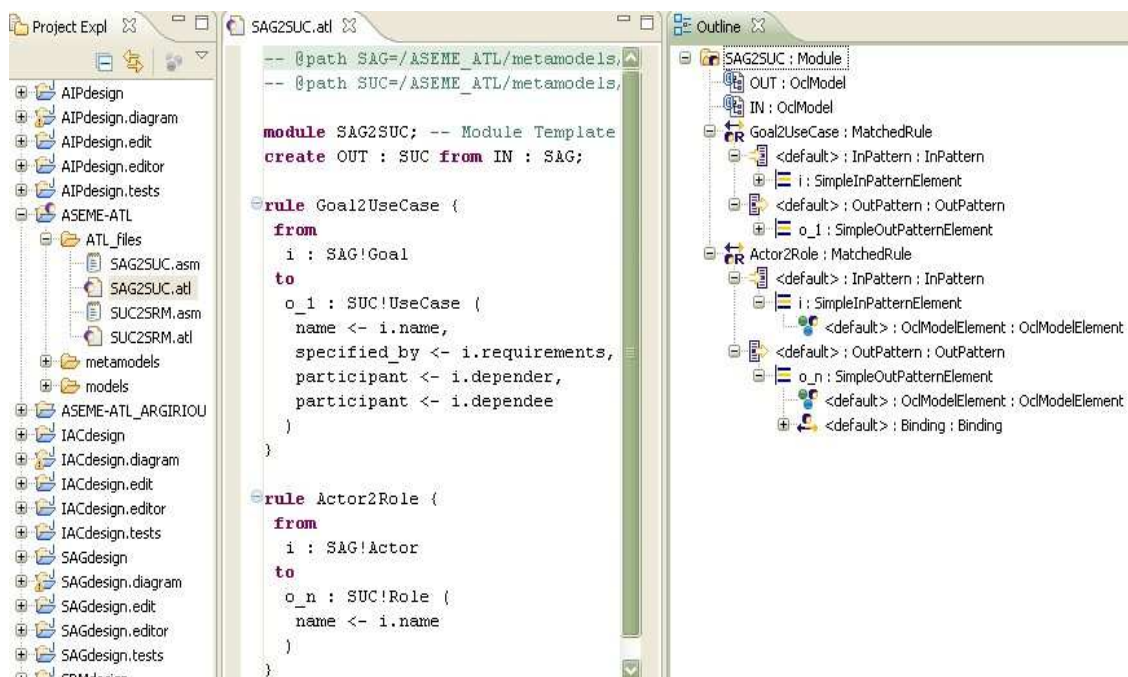
Η δομή ενός project ATL μετασχηματισμού στο Ολοκληρωμένο Περιβάλλον Ανάπτυξης Eclipse (IDE Eclipse) φαίνεται στην εικ. 14. Τρεις φάκελοι περιέχουν τα χρησιμοποιούμενα αρχεία , (ο φάκελος *ATL_files* περιέχει τα scripts μετασχηματισμού στην ATL , ο φάκελος *metamodels* περιέχει τα μεταμοντέλα και ο φάκελος *model* περιέχει τα μοντέλα.

SAG2SUC Μετασχηματισμός

Ο SAG2SUC μετασχηματισμός είναι ένας κλασσικός M2M μετασχηματισμός χρησιμοποιώντας ATL. Στην κορυφή, τα IN και OUT μεταμοντέλα ορίστηκαν ακολουθούμενα από κανόνες που έχουν σαν στιγμιότυπα εισόδου και εξόδου, αντίστοιχα μοντέλα. Ο πρώτος κανόνας (*Goal2UseCase*) παίρνει ως είσοδο ένα σχέδιο SAG στόχο και δημιουργεί ένα σχέδιο SUC περίπτωσης χρήσης αντιγράφοντας τις ιδιότητές του. Η ATL γλώσσα μετασχηματισμού είναι *δηλωτική* και έχει δημιουργηθεί για τις περιπτώσεις όπου ένα σχέδιο αναφέρει κάποιο άλλο. Οι αναφορές *depender* και



dependee ενός SAG Goal μετασχηματίσθηκαν αμφότερες σε *interacter* (ή *participant*) αναφορές (EReferences) του SUC UseCase. Η ATL μηχανή εννοεί ότι ο μετασχηματισμός δεν αφορά ένα EAttribute (όπως στην περίπτωση του *attribute name*) και ψάχνει τους κανόνες για να βρει έναν που θα μετασχηματίσει τους τύπους της EReference (π.χ. ο SAG Actor στον SUC Role). Βρίσκει τον δεύτερο κανόνα (Actor2Role) και τον ενεργοποιεί, δημιουργώντας έτσι τα αντικείμενα τύπου EReference και ολοκληρώνοντας την ενεργοποίηση του πρώτου κανόνα (εικ. 14).



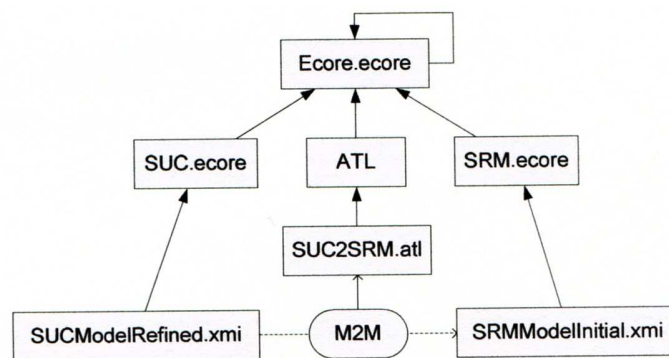
Εικ. 14. Η δομή ενός project ATL μετασχηματισμού στο Ολοκληρωμένο Περιβάλλον Ανάπτυξης Eclipse (IDE Eclipse)

SUC2SRM Μετασχηματισμός

Μετά τον μετασχηματισμό SAG2SUC ο σχεδιαστής εργάζεται με το μοντέλο SUC (χρησιμοποιώντας τον *Sample Reflective Ecore Model Editor* του Eclipse) προσθέτοντας λεπτομέρεια στον τύπο των *included use cases*, χρησιμοποιώντας ειδικές τεχνολογίες στον ορισμό των use cases και της πολλαπλότητας. Όταν τελειώσει μπορεί να προχωρήσει στον επόμενο μετασχηματισμό SUC2SRM. Η ATL τεχνολογία χρησιμοποιείται πάλι και οι κανόνες για αυτόν τον μετασχηματισμό παρουσιάζονται



στην εικ.15.Έχει έναν επιπλέον κανόνα καθώς οι UseCases μπορούν να μετασχηματισθούν είτε σε Capabilities (αν αυτές κάνουν *include* άλλες) ή σε απλές activities.Αυτή την φορά πάλι, ο πρώτος κανόνας (Role2Role) ενορχηστρώνει την διαδικασία μετασχηματισμού.Το σχήμα μετασχηματισμού παρουσιάζεται στην εικ. 15.Οι κανόνες ATL εφαρμόζονται στο εξευγενισμένο SUC μοντέλο(SUCModelRefined.xmi) και δημιουργείται το αρχικό SRM μοντέλο (SRMModelInitial.xmi).Ο ASEME σχεδιαστής έχει τώρα να εκτελέσει τον ASEME_SUC2SRM ATL μετασχηματισμό.



Εικ. 15 .Το σχήμα του SUC2SRM M2M μετασχηματισμού

2.3 Χρήση Γραφικών Εργαλείων σε Υπάρχουσες Μεθοδολογίες

Η Πρακτοροστραφής Μηχανική Λογισμικού (Agent Oriented Software Engineering-AOSE) αναδύεται, μετά την εγκαθίδρυση των αυτόνομων πρακτόρων και των συστημάτων πολλαπλών πρακτόρων (Multi-Agent Systems - MAS) , ως ένα ερευνητικό πεδίο της επιστήμης των υπολογιστών ή αυστηρότερα της τεχνητής νοημοσύνης.Η πρώτη εργασία με αυτό το όνομα έγινε το 2001 [6] , αν και ο όρος είχε ήδη εμφανισθεί σε προηγούμενες εργασίες.

Η παρακτοροστραφής ανάπτυξη διαφαίνεται ως ένα επόμενο βήμα στην εξέλιξη της μηχανικής ανάπτυξης λογισμικού.Απο τον παραπάνω λοιπόν χώρο δημιουργήθηκε η ανάγκη για την ανάπτυξη λογισμικού το οποίο θα είναι [39] , [38] :

- *Αυτόνομο (autonomous)*.Οι πράκτορες μπορούν να λειτουργήσουν χωρίς την απευθείας ανθρώπινη επέμβαση ή άλλων οντοτήτων και μπορούν να έχουν



κάποιο είδος ελέγχου των δράσεων , της εσωτερικής κατάστασης και της κατανάλωσης των πόρων τους.

- *Κοινωνικό (social)*.Οι πράκτορες χρησιμοποιούν μια γλώσσα επικοινωνίας για να αλληλεπιδράσουν με άλλους πράκτορες (και πιθανόν ανθρώπους).Έχουν κάποιο είδος ελέγχου των απαλλαγών τους και μπορούν να επιλέξουν τους συνεργάτες τους για την επίλυση προβλημάτων.
- *Αντιδραστικό (reactive)*.Οι πράκτορες αντιλαμβάνονται το περιβάλλον τους και ανταποκρίνονται την κατάλληλη χρονική στιγμή στις αλλαγές που συμβαίνουν σ' αυτό , σύμφωνα με τους στόχους τους.
- *Ικανό να παίρνει πρωτοβουλίες (pro-active)*.Οι πράκτορες μπορούν να επιδεικνύουν την κατευθυνόμενη από τους στόχους συμπεριφορά τους παίρνοντας πρωτοβουλία , γίνονται σκόπιμοι , μη αντιδρώντας απλά στις αλλαγές του περιβάλλοντος.
- *Προσαρμοστικό (adaptive)*.Οι πράκτορες μπορούν να προσαρμόζονται στις αλλαγές του περιβάλλοντός τους.
- *Διαρκές (persistent)*.Οι πράκτορες έχουν μια μεγάλη διάρκεια , αντίθετα με τα αντικείμενα που στιγμιτυποποιούνται να κάνουν κάτι και μετά στέλνονται στα σκουπίδια.

Τα συστήματα που συντίθενται από αλληλεπιδραστικούς πράκτορες ονομάζονται συστήματα πολλαπλών πρακτόρων (Multi-Agent Systems-MAS).Τα νέα χαρακτηριστικά και οι ιδέες των πρακτόρων αυτών χρειάζονται να ολοκληρωθούν σε ένα λογισμικό μιας μεθοδολογίας μηχανικής.Η Πρακτοροστραφής ανάπτυξη λογισμικού (Agent Oriented System Engineering-ASOE) έρχεται να καλύψει την ανάγκη συνδυασμού τμημάτων από διαφορετικές μεθοδολογίες , τα οποία είναι επαναχρησιμοποιούμενα μεθοδολογικά τμήματα που μπορούν να χρησιμοποιήσουν οι μηχανικοί για να παράγουν μια νέα διεργασία σχεδιασμού για μια ιδιαίτερη κατάσταση.Αυτό επιτρέπει στην ομάδα ανάπτυξης να έρθει με μια υβριδική μεθοδολογία που θα υποστηρίζει τις ανάγκες ενός ειδικού προγραμματισμού και τις ικανότητες μοντελοποίησης.

Ένα από τα σημαντικότερα ζητήματα στην μεθοδολογία αυτή είναι η μοντελοποίηση, η παρουσίαση και η υλοποίηση των πρωτοκόλλων αλληλεπίδρασης πρακτόρων.Μια



ευρείας έκτασης ομάδα μεθοδολογιών για την AOSE είτε προσαρμόζουν ένα υπάρχον μοντέλο (AUML) , είτε χειρίζονται μοντέλα UML ή απλά ορίζουν μηνύματα που οι πράκτορες στέλνουν ο ένας στον άλλο , επιτρέποντας την μοντελοποίηση απλών πρωτοκόλλων.

Ακολούθως θα αναφερθούμε , για τις ανάγκες της παρούσας εργασίας σε δύο απ' τις ASOE μεθοδολογίες.

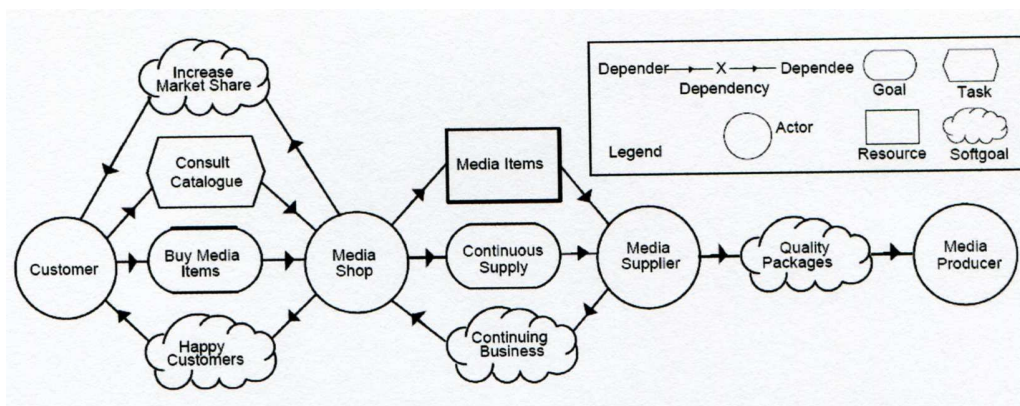
2.3.1 Μεθοδολογία Tropos

Πολλές μεθοδολογίες μηχανικής πρακτοροστραφούς λογισμικού έχουν προταθεί για να υποστηρίξουν την ανάπτυξη σύνθετων διανεμόμενων συστημάτων που περιλαμβάνουν συστήματα πολλαπλών πρακτόρων (MAS).

Μία απ' αυτές είναι η μεθοδολογία ανάπτυξης πρακτοροστραφούς λογισμικού *Tropos* που προτείνει την χρήση δραστών , στόχων, σχεδίων και εξαρτήσεων μεταξύ δραστών σε όλη την πορεία ανάπτυξης λογισμικού. Η διεργασία καλύπτει πέντε φάσεις ανάπτυξης : α) την ανάλυση των πρώιμων απαιτήσεων , β) την ανάλυση των έσχατων απαιτήσεων , γ) τον αρχιτεκτονικό σχεδιασμό , δ) τον λεπτομερή σχεδιασμό και ε) την υλοποίηση. Ο *Tropos* χρησιμοποιεί μια γλώσσα οπτικής παρουσίασης βασισμένη στο i* framework. Έτσι ένα μοντέλο *Tropos* μπορεί να παρουσιασθεί ως ένα σύνολο από διαγράμματα όπως για παράδειγμα τα διαγράμματα δραστών που περιγράφουν το δίκτυο των σχέσεων εξάρτησης μεταξύ των δραστών , διαγράμματα στόχων που εμφανίζουν ανάλυση στόχων και σχεδίων από την άποψη του συγκεκριμένου στόχου. Οι στόχοι διαιρούνται σε δύο κατηγορίες , τους σκληρούς (που αναφέρονται στις λειτουργικές ιδιότητες των δραστών (και τους μαλακούς (που αναφέρονται στις μη λειτουργικές ιδιότητες των δραστών). Έτσι τα διαγράμματα δραστών αναλύουν τους στόχους ενός συγκεκριμένου δράστη σε υποστόχους και σχέδια για επίτευξη του στόχου. Στην φάση των τελευταίων απαιτήσεων τα μοντέλα επεκτείνονται προσθέτοντας πιθανές αλληλεπιδράσεις μεταξύ των δραστών (χρήσιμοι ή συγκρουόμενοι στόχοι). Ένα απλό διάγραμμα δραστών παρουσιάζεται στην εικ 16. για ένα media market. Οι κύριοι δράστες είναι οι Customer, Media Shop , Media Supplier και Media Producer. Ο δράστης Customer εξαρτάται από τον δράστη Media Shop για να εκπληρώσει τον στόχο "Buy Media Items". Ο δράστης Media Shop εξαρτάται από τον δράστη Customer για τους μαλακούς στόχους του



“Increase Market Share” και “Happy Customers”.Ο Customer επίσης εξαρτάται από τον Media Shop για να εκπληρώσει την διεργασία “Consult Catalogue”.Όμοια, υπάρχουν εξαρτήσεις μεταξύ των δραστών Media Shop και Media Supplier και μεταξύ των δραστών Media Supplier και Media Producer για να συμπληρώσουν την αλυσίδα.Στην φάση της ανάλυσης των τελευταίων απαιτήσεων βέβαια οι επιλεγμένοι για υλοποίηση δράστες , σχέδια και στόχοι ραφινάρονται.



Εικ. 16. Διάγραμμα δραστών για ένα media market [17]

Το περιβάλλον βασίζεται στην Πλατφόρμα Ανάπτυξης Eclipse που προσφέρει μια ευέλικτη λύση στο πρόβλημα της ολοκλήρωσης των συστατικών.Νέα εργαλεία μπορούν να ολοκληρωθούν στην πλατφόρμα μέσω plug-ins , την μικρότερη λειτουργική μονάδα στο Eclipse , που παρέχουν νέες λειτουργικότητες στο περιβάλλον.

Το τελικό περιβάλλον συντίθεται από διαφορετικά εργαλεία που υποστηρίζουν τον αναλυτή στα διαφορετικά βήματα της μεθοδολογίας , όπως φαίνεται στην εικ. 17 , όπου δίνεται μια σκιαγραφία της χρήσης του περιβάλλοντος παράλληλα με τις πέντε φάσεις ανάπτυξης.

Φάση ανάλυσης πρώιμων απαιτήσεων.Εδώ εκτελείται ο προσδιορισμός του αρχικού μοντέλου από το εργαλείο μοντελοποίησης TAOM4E (Tool for Agent Oriented Modeling for Eclipse)(ετικέττα-1) που έχει αναπτυχθεί μέσα στο Eclipse , στην κορυφή δύο υπάρχοντων plug-ins , του GEF(Graphical Editing Framework) plug-in που διαχειρίζεται τις λειτουργίες σύνταξης και του EMF plug-in που προσφέρει ένα περιβάλλον μοντελοποίησης και παραγωγή κώδικα για τα συμμορφούμενα στο MOF μεταμοντέλα.



Το TAOM4E συνίσταται από δύο plug-ins : α) το μοντέλο TAOM4E , που υλοποιεί το Tropos μεταμοντέλο εκμεταλλευόμενοι το EMF plug-in και β) την πλατφόρμα TAOM4E , που υλοποιεί τις συναρτήσεις του σχεδιαστή για δημιουργία και διαχείριση ενός μοντέλου Tropos εκμεταλλευόμενοι το GEF plug-in. Το εργαλείο μοντελοποίησης υποστηρίζει τον προσδιορισμό του μοντέλου Tropos μέσω Γραφικής Διασύνδεσης Χρήστη (Graphical Unit Interface-GUI), καθώς επίσης επιτρέπει να αποθηκεύονται τα Tropos μοντέλα σε XMI μορφή.

Φάση εσχάτων απαιτήσεων. Και αυτή η φάση υποστηρίζεται από το εργαλείο TAOM4E. Εδώ το προς δημιουργία σύστημα εισάγεται στο domain μοντέλο ως ένας νέος δράστης. Οι εξαρτήσεις ανάμεσα στο σύστημα και τους άλλους δράστες στο domain προσδιορίζονται λεπτομερώς και εκτελείται μια ανάλυση εσωτερικού συστήματος ,μέσω τεχνικών ανάλυσης στόχων. Με τον τρόπο αυτό προσδιορίζονται λεπτομερώς οι απαιτήσεις του συστήματος.

Φάση αρχιτεκτονικού σχεδιασμού. Στην φάση αυτή το TAOM4E επιτρέπει να προσδιορισθούν τα υποσυστήματα του προς δημιουργία συστήματος και οι σχέσεις τους με άλλα υποσυστήματα. Επιπλέον , εδώ , προσδιορίζεται το σύνολο των ικανοτήτων (capabilities) του συστήματος μέσω της σχέσεων δράστη/σχεδίου με συνεισφορά των σχέσεων σχεδίου/μαλακών στόχων. Το σύνολο των ικανοτήτων μετά διατηρείται σε ένα πίνακα ικανοτήτων.

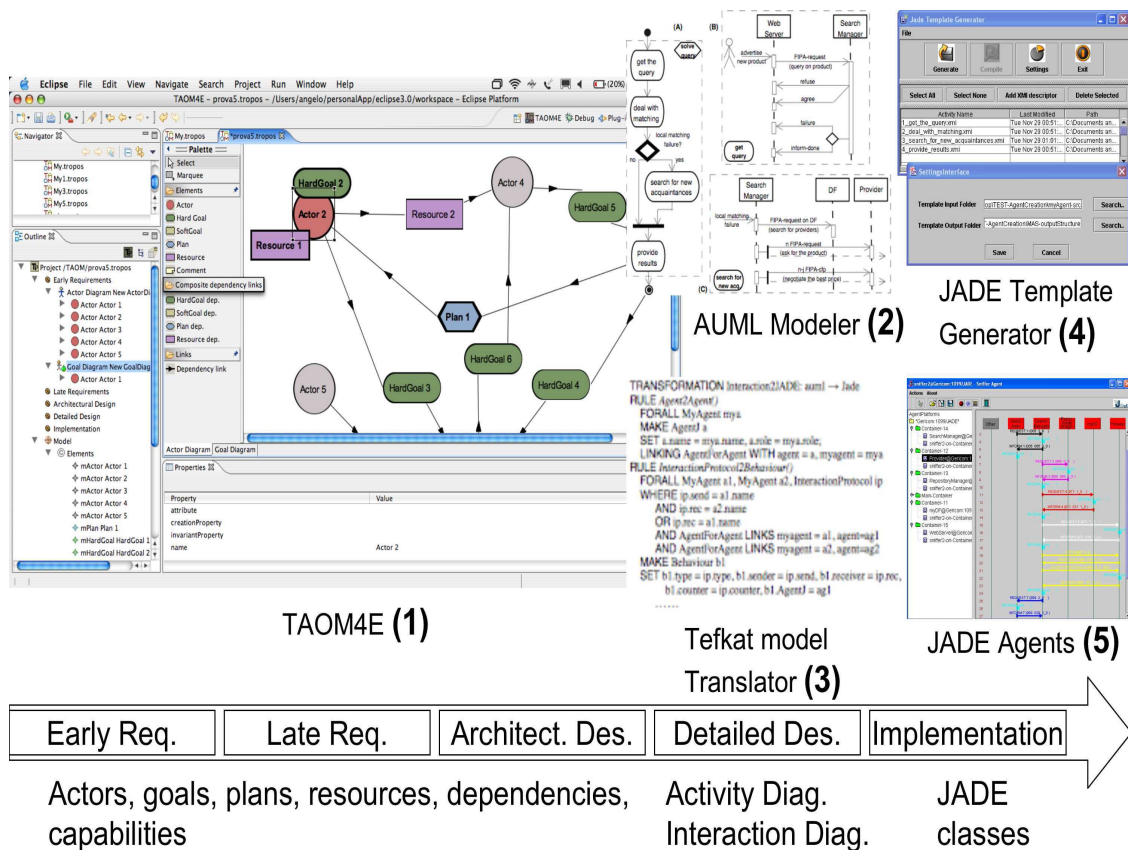
Φάση λεπτομερούς σχεδιασμού. Σ' αυτό το στάδιο , χρησιμοποιούμε UML/AUML ικανότητα και διαγράμματα για να μοντελοποιήσουμε μέρος των δυναμικών ιδιοτήτων των ικανοτήτων των πρακτόρων που προσδιορίστηκαν στην προηγούμενη φάση. Προτείνουμε την χρήση τεχνικών μετασχηματισμού στο επίπεδο μεταμοντέλου , που αναφέρονται στην QVT. Ιδιαίτερα χρησιμοποιείται ένα plug-in μετασχηματισμού του Eclipse , ως μηχανή μετασχηματισμού μοντέλου που λέγεται Tefkat(ετικέττα-3 και υλοποιεί ένα σύνολο των απαιτήσεων. Χρησιμοποιούμε την QVT για να μετασχηματίσουμε το μοντέλο των ικανοτήτων των πρακτόρων , που εκφράζονται μέσω των οντοτήτων του Tropos και κατασκευών που μπορεί να γίνουν μέσω ενός UML/AUML σχεδιαστή (ετικέττα-2). Στην ίδια φάση, εκτελείται και ένας δεύτερος μετασχηματισμός για να μετασχηματίσει σε επίπεδο μεταμοντέλου την ικανότητα και τα



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

διαγράμματα ακολουθίας που προσδιορίζουν την συμπεριφορά του πράκτορα , στο σύνολο των κλάσεων που παρουσιάζονται στο μεταμοντέλο JADE.

Φάση υλοποίησης. Κατά την διάρκεια της φάσης αυτής , ο τελευταίος μετασχηματισμός εκτελείται για να παράγει τις JADE κλάσεις του συνόλου των πρακτόρων , αρχίζοντας από τις JADE κλάσεις που ορίστηκαν κατά την διάρκεια του τελευταίου τμήματος του λεπτομερούς σχεδιασμού. Αυτός ο μετασχηματισμός εκτελείται μέσω του εργαλείου *JADE Template Generator* (ετικέττα-4) που αρχίζει από τον προσδιορισμό των ικανοτήτων του πράκτορα (σε XMI αρχεία) , παραγόμενες αυτόματα από την μηχανή Tefkat και παράγει JADE κώδικα πρακτόρων (ετικέττα-5)



Εικ. 17. Η 5 φάσεων διαδικασία της Tropos και το περιβάλλον ανάπτυξης οδηγούμενο από μοντέλα που υποστηρίζει την μεθοδολογία

Οι απαιτήσεις για την έκδοση 0.6.3 του εργαλείου TAOM4E και τις επεκτάσεις του (εκτός αν προσδιορίζονται διαφορετικά) είναι :



- Sun JAVA jdk 6.0
- Eclipse >3.4
- EMF (Eclipse Modeling Framework)κατάλληλο για την εγκατεστημένη έκδοση του Eclipse
- GEF (Graphical Editing Framework) κατάλληλο για την εγκατεστημένη έκδοση του Eclipse

Τα EMF και GEF εγκαθίστανται αυτόματα όταν εγκαθίσταται το TAOM4E μέσω του διαχειριστή ενημερώσεων του Eclipse. Διφορετικά , εγκαθίστανται κατευθείαν χρησιμοποιώντας τον εν λόγω διαχειριστή. Κατεβάζουμε το συμπίεμένο αρχείο και το αποσυμπιέζουμε παίρνοντας δύο φακέλους με ονόματα `it.itc.taom4e.model_0.5.0` και `it.itc.taom4e.platform_0.5.0` , στον φάκελο `eclipse/plugins`.

Το TAOM4E είναι ελεύθερο λογισμικό που διανέμεται κάτω από την άδεια GPL_v2. Το `t2x` εργαλείο είναι ολοκληρωμένο στο Taom4e TROPOS plug-in μοντελοποίησης του Eclipse και εγκαθίσταται μαζί του από το site ενημέρωσής του :

<http://sra.itc.it/tools/taom4e/eu.fbk.se.taom4e.updateSite> (για περισσότερες πληροφορίες : <http://disi.unitn.it/~morandini>)

2.3.2 Μεθοδολογία Ingenias

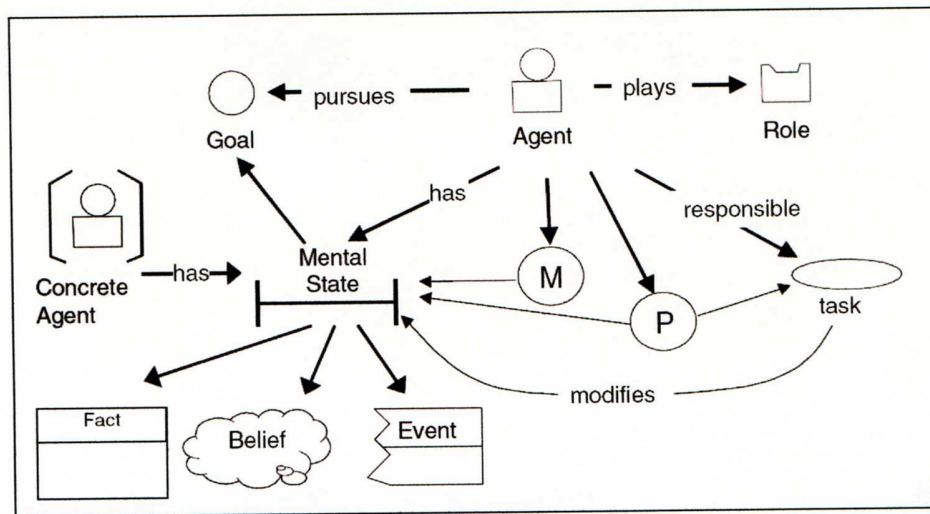
Igenias [29] είναι μια μεθοδολογία που αναδύεται με ένα περιβάλλον ανάπτυξης επιτρέποντας την ανάπτυξη πρακτόρων χρησιμοποιώντας το μεταμοντέλο Ingenias. Το μεταμοντέλο του είναι το πλουσιότερο ανάμεσα στις AOSE μεθοδολογίες περιέχοντας περισσότερα από 300 σχέδια (το `ecore` μεταμοντέλο του Ingenias μπορεί να «κατεβασθεί» από το διαδίκτυο <http://igenias.sourceforge.net>). Αυτό το χαρακτηριστικό μπορεί επίσης να θεωρηθεί ως εξαιρετικά περιοριστικό για τους σχεδιαστές που θέλουν να χρησιμοποιήσουν τις δικές τους αρχιτεκτονικές πρακτόρων , αλλά επίσης χρειάζονται περισσότερο χρόνο εκμάθησης από ότι όλες οι άλλες μεθοδολογίες για να αρχίσουν να δουλεύουν με αυτήν. Η μέθοδός της είναι επίσης δύσκολη στην εκμάθηση και χρήση (ιδιαίτερα στην αναδρομική ανάπτυξη) καθώς συνίσταται από πάνω από 100 δραστηριότητες περίπου [30] .



Ορίζει ένα ολόκληρο καινούργιο σύνολο από μοντέλα και τα συνδέει με τα UML μοντέλα σκοπεύοντας να ορίσει τα σχέδια τα σχετικά με την ανάπτυξη πρακτόρων και την εγκατάστασή τους στην UML , βοηθώντας την φάση ανάπτυξης με μια αντικειμενοστραφή γλώσσα.Ο αναγνώστης μπορεί να πάρει μόνο μια γεύση των διαγραμμάτων IGNEIAS στην εικ.18 (στην εργασία αυτή δεν θα μπούμε σε λεπτομέρειες αυτής της μεθοδολογίας καθώς θα ήταν πολύ μακροσκελής).Η λειτουργικότητα ενός πράκτορα περιγράφεται με όρους στόχων , διεργασιών και ικανοτήτων (ή ρόλων που αυτός παίζει).Αυτά καλύπτονται από τους ακόλουθους όρους:

- *Η νοητική κατάσταση (The Mental State)* περιλαμβάνει όλες τις πληροφορίες που χρειάζονται για την λήψη απόφασης της πορείας ενός πράκτορα.Αυτές οι πληροφορίες είναι οι στόχοι , οι πεποιθήσεις και τα γεγονότα του πράκτορα.
- *Ο διαχειριστής διανοητικής κατάστασης (M)(The Mental State Manager)* παρέχει λειτουργίες για την δημιουργία , την διαγραφή και την τροποποίηση οντοτήτων διανοητικής κατάστασης.
- *Ο επεξεργαστής διανοητικής κατάστασης (The Mental State Processor)* είναι υπεύθυνος να αποφασίζει ποιά διεργασία να εκτελεσθεί μεταξύ των διεργασιών του πράκτορα.

Η Ignenias διακρίνει καθαρά ανάμεσα σε έναν πράκτορα και σε μια εφαρμογή δείχνοντας ότι η τεχνολογία πρακτόρων υπάρχει για να προσθέτει νέα χαρακτηριστικά στα συστήματα υπολογιστών.Οι πράκτορες έχουν πρόσβαση στις εφαρμογές μέσω ενός είδους προγραμματισμένης διασύνδεσης εφαρμογών (Application Programming Interface-API) που αυτοί προσφέρουν.Ένα ζήτημα που η IGNEIAS αφήνει στον σχεδιαστή είναι αν αυτός θα ορίσει πρώτα τις διεργασίες ή τους στόχους ενός πράκτορα.Ίσως αυτό είναι το αποτέλεσμα της έλλειψης της φάσης της ανάλυσης απαιτήσεων.Επιπλέον , η Ignenias δεν προσφέρει την ευκολία της βαθμιαίας μοντελοποίησης ενός συστήματος πολυπρακτόρων μελετώντας το σε διαφορετικά επίπεδα αφάιρεσης.



Εικ. 18. Στοιχεία της άποψης πράκτορα στην IGNENIAS

Οι Garcia-Magarino et. al. (2009) [11] στην πρωτότυπη εργασία τους παρουσίασαν έναν αλγόριθμο που παράγει μετασχηματισμούς μοντέλων από παράδειγμα. Αυτός ο αλγόριθμος ευκολύνει την παραγωγή των πολλοί -προς -πολλοί μετασχηματισμών μεταξύ των αυθαίρετων διαγραμμάτων των στοιχείων, συμφωνώντας με τις γλώσσες μετασχηματισμού να μην υποστηρίζουν απ' ευθείας διαγράμματα των στοιχείων στα μοντέλα πηγής και στόχου αντίστοιχα. Αυτοί ανέπτυξαν το εργαλείο MTGenerator για την εφαρμογή του αλγόριθμου να υποστηρίζει τις διαδικασίες αντικειμενοστρεφούς λογισμικού της μεθοδολογίας IGNENIAS, η οποία υλοποιεί τον αλγόριθμο για την γλώσσα μετασχηματισμού ATLAS.

Η προσέγγισή τους επιτρέπει στον μηχανικό να ορίσει μόνος του τους μετασχηματισμούς που θέλει να εφαρμόσει στα μοντέλα που συμμορφώνονται με το IGNENIAS μεταμοντέλο.

Λαμβάνοντας υπ' όψιν το τεράστιο IGNENIAS μεταμοντέλο και τα πολλά πιθανά μονοπάτια που ο μηχανικός μπορεί να ακολουθήσει, αυτή η λύση από την μια πλευρά δίνει ελευθερία στον μηχανικό, αλλά από την άλλη πλευρά τον καίει με την επιπλέον δουλειά να ορίσει μόνος του τους μετασχηματισμούς.

Το εργαλείο ανάπτυξης λογισμικού IGNENIAS (IDK-Ignenias Development Kit)



(<http://grasia.fdi.ucm.es>)

- υποστηρίζει την IGNENIAS μεθοδολογία (Ανάπτυξη Οδηγούμενη από Μοντέλα)
- επιτρέπει οπτικό προσδιορισμό της MAS
- είναι αναπτυγμένη σε java. , πολλαπλών πλατφορμών και ανοικτού κώδικα.Ακολουθεί GPL (General Public Licence) 2.0
- είναι πάντα ενημερωμένη και πάντα συναφής με το IGNENIAS μεταμοντέλο

ενώ τα κύρια χαρακτηριστικά του είναι :

- αυτόματη παραγωγή κώδικα για το JADE INGENIAS Agent Framework
- παραγωγή HTML (HyperText Markup Language)
- ευκολίες εκσφαλμάτωσης
- προσαρμοσμένες επεκτάσεις
- εκτελέσιμη με αυτόνομο τρόπο

2.4 Εργαλεία Γραφικής Αναπαράστασης μοντέλων

2.4.1 GMF (Graphical Modeling Framework)-Eclipse Modeling Framework (EMF)

Το πλαίσιο γραφικής μοντελοποίησης (Graphical Modeling Framework - GMF), ένα πρόγραμμα γραφικής μοντελοποίησης του Eclipse , έχει σκοπό να παράγει μια παραγωγική γέφυρα μεταξύ του πλαισίου μοντελοποίησης του Eclipse (Eclipse Modeling Framework-EMF) και του πλαισίου γραφικής έκδοσης (Graphical Editing Framework - GEF) , δηλ. παρέχει ένα σύνολο από παραγωγικά συστατικά και εκτελέσιμες υποδομές , π.χ. διαγράμματα κλάσεων , για ανάπτυξη γραφικών συντακτών βασισμένων στα EMF και GMF.

Πριν προχωρήσουμε στην περιγραφή και χρήση του GMF , ας αναφερθούμε για λίγο στα χαρακτηριστικά των επί μέρους πλαισίων , EMF και GEF , αντίστοιχα.

Eclipse Modeling Framework (EMF)

Είναι μια ευκολία πλαισίου μοντελοποίησης και παραγωγής κώδικα για κατασκευή εργαλείων και άλλων εφαρμογών βασισμένων ένα δομημένο πρότυπο δεδομένων.Απο ένα προσδιορισμό προτύπου που περιγράφεται σε XMI , το EMF παρέχει εργαλεία και



υποστήριξη για παραγωγή ενός συνόλου Java κλάσεων για το μοντέλο , ενός συνόλου κλάσεων προσαρμοστών που κάνουν δυνατή την εμφάνιση και την βασισμένη σε εντολές σύνταξη του μοντέλου.Το σημαντικότερο από όλα είναι ότι το EMF παρέχει την δυνατότητα για διαλειτουργικότητα με άλλα , βασισμένα στο EMF , εργαλεία και εφαρμογές.Επίσης περιλαμβάνει τα υποπρογράμματα καθορισμός σχημάτων XML (XML Schema Definition-XSD) , ενός συστατικού του προγράμματος εργαλείων ανάπτυξης μοντέλων (Model Development tools – MDT) και μια βασισμένη στο EMF υλοποίηση του προγράμματος των αντικειμένων υπηρεσιών δεδομένων (Service Data Objects – SDO).

EMF & Ecore

- Ecore : είναι μια βασισμένη στο Eclipse υλοποίηση του MOF
- EMF παρέχει αυτόματη δημιουργία συντακτών με απλή δένδροειδή μορφή από Ecore διαγράμματα
- Ο κώδικας από τον βασισμένο στον EMF συντάκτη μπορεί να επεκταθεί και διαμορφωθεί από τον σχεδιαστή

Η βήμα προς βήμα δημιουργία ενός Ecore μοντέλου

- Έναρξη του Eclipse
- Δημιουργία νέου άδειου EMF project
- Δημιουργία Ecore αρχείου
- Ονομασία του πακέτου , αποθήκευση του αρχείου
- Δημιουργία εμφάνισης διαγράμματος
- Δημιουργία κλάσεων , ιδιοτήτων και συνδέσμων
- Εξέταση ορθότητας τύπων και πολλαπλοτήτων

Κατασκευή ενός απλού συντάκτη

- Δημιουργία του EMF παραγωγού μοντέλου (generator model) και παραγωγή κώδικα (generate code)
- Δυνατή χρησιμοποίηση αμφοτέρων, του δυναμικού συντάκτη στον πάγκο εργασίας (workbench) , ή έναρξη νέου εκτελέσιμου πάγκου εργασίας για έλεγχο του συντάκτη



Παρακάτω παρουσιάζεται ένα διάγραμμα που επεξηγεί τα κύρια συστατικά και τα μοντέλα που χρησιμοποιούνται κατά την διάρκεια της βασισμένης στο GMF ανάπτυξης. Πυρήνας στο GMF είναι η έννοια του μοντέλου του γραφικού καθορισμού, το οποίο μοντέλο περιέχει πληροφορίες για τα γραφικά στοιχεία που θα εμφανισθούν σε μια βασισμένη στο GEF εκτέλεση, αλλά δεν έχουν απευθείας σύνδεση με τα domain μοντέλα για τα οποία θα παράξουν την παρουσίαση και την σύνταξή τους. Ένας επιλεκτικός καθορισμός σχεδίασης χρησιμοποιείται για να σχεδιάσει την παλέτα και τα άλλα περιφερειακά (μενού, μπάρες εργαλείων κλπ).

Αναμένεται ο γραφικός καθορισμός ή ο καθορισμός σχεδίασης μπορεί να δουλεύει το ίδιο καλά για αρκετά domain μοντέλα. Για παράδειγμα, το διάγραμμα κλάσεων της UML έχει πολλά αντίστοιχα, τα οποία είναι καταπληκτικά παρόμοια στην βασική τους εμφάνιση και δομή. Ένας στόχος του GMF είναι να επιτρέπει στον γραφικό καθορισμό να μπορεί να επαναχρησιμοποιηθεί για αρκετά domain μοντέλα. Αυτό επιτυγχάνεται με την χρήση ενός μοντέλου χαρτογράφησης που συνδέει τον γραφικό καθορισμό και τον καθορισμό σχεδίασης με το επιλεγμένο (ή τα επιλεγμένα) μοντέλο (ή μοντέλα).

Μόλις καθορισθούν οι κατάλληλες χαρτογραφήσεις, το GMF παρέχει μια γεννήτρια μοντέλου που επιτρέπει να ορισθούν οι λεπτομέρειες υλοποίησης για την φάση της παραγωγής. Η παραγωγή ενός plug-in του συντάκτη με την γεννήτρια μοντέλου θα οδηγήσει στο τελικό μοντέλο, που είναι το διάγραμμα χρόνου εκτέλεσης. Ο χρόνος εκτέλεσης θα γεφυρώσει το διάγραμμα χρόνου εκτέλεσης με το/τα domain μοντέλο/μοντέλα, όταν ένας χρήστης δουλεύει με ένα διάγραμμα, παρέχοντας επίσης διάρκεια και συγχρονισμό των δύο. Μια σπουδαία πτυχή αυτού του χρόνου εκτέλεσης είναι ότι παρέχει μια προσέγγιση βασισμένη στις υπηρεσίες της λειτουργικότητας των EMF και GEF και είναι σε θέση να ενεργοποιηθούν από μη-παραγόμενες εφαρμογές.

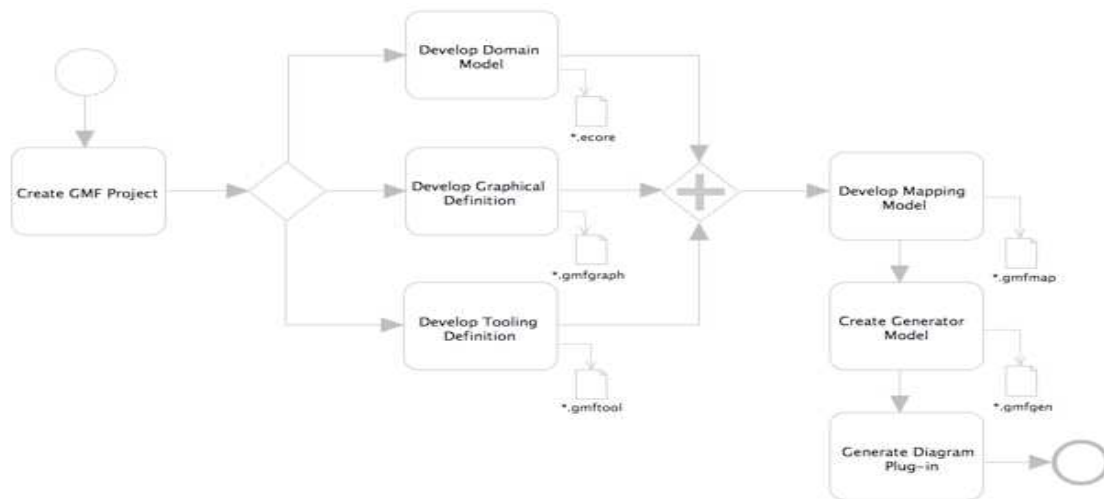
Αφού καλύφθηκαν τα βασικά του GMF, θα αναφερθούμε στην διαδικασία χρησιμοποίησης του GMF στην ανάπτυξη ενός γραφικού συντάκτη επιφάνειας, για ένα παράδειγμα domain μοντέλου, παρουσιάζοντας τα βήματα δημιουργίας ενός βασισμένου στο GMF γραφικού συντάκτη, δίνοντας χαρακτηριστικές εικόνες από την διαδικασία αυτή [8,14].

Η βήμα προς βήμα δημιουργία ενός γραφικού συντάκτη μοντέλου

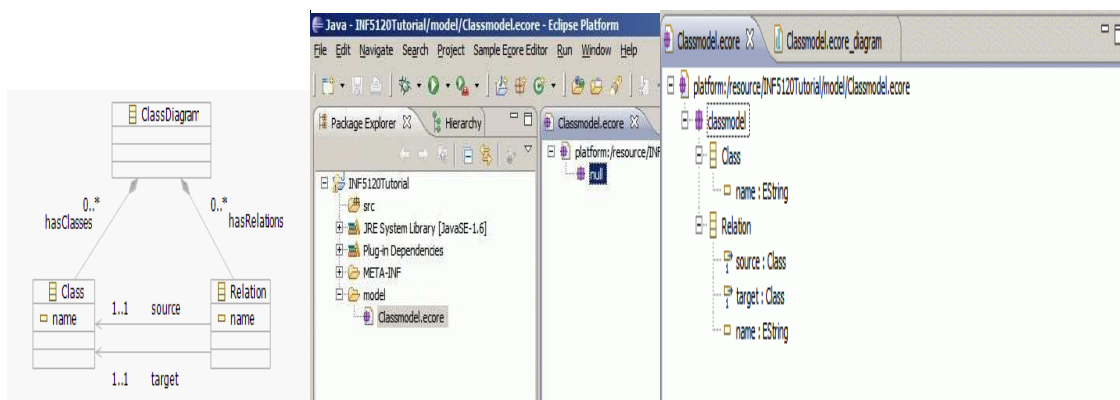


Αργυρίου Αντώνιος, Έμετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

- Άνοιγμα του Eclipse
- Εγκατάσταση του GMF και των εξαρτήσεών του , μέσω του Eclipse
- Δημιουργία νέου GMF project
- Καθορισμός (σχεδιασμός) του επιλεγμένου domain μοντέλου
- Γραφικός καθορισμός μοντέλου (graphical definition)
- Καθορισμός σχεδίασης μοντέλου (tooling definition)
- Καθορισμός χαρτογράφησης μοντέλου (mapping definition)
- Παραγωγή κώδικα (code generation)
- Τρέξιμο διαγράμματος (running the diagram)



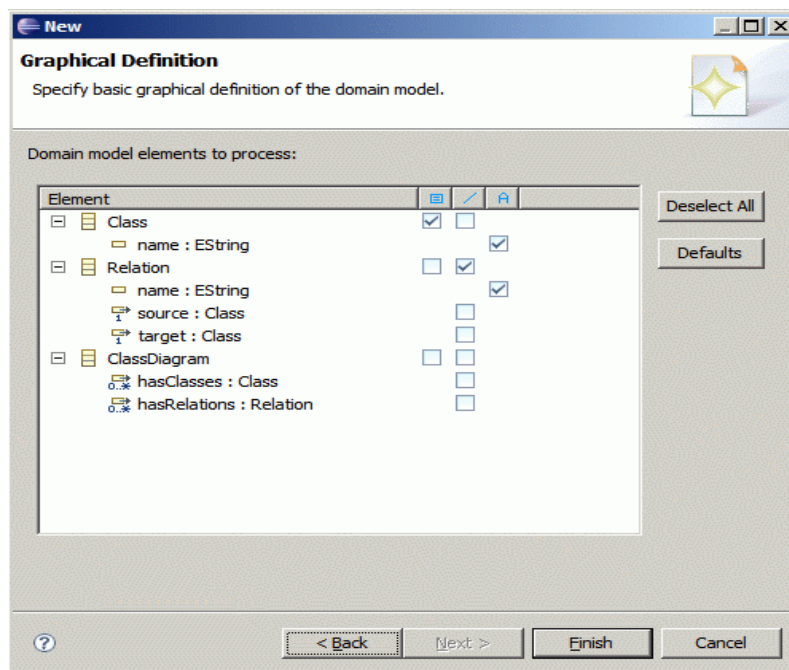
Εικ. 19. Επισκόπηση του πλαισίου GMF



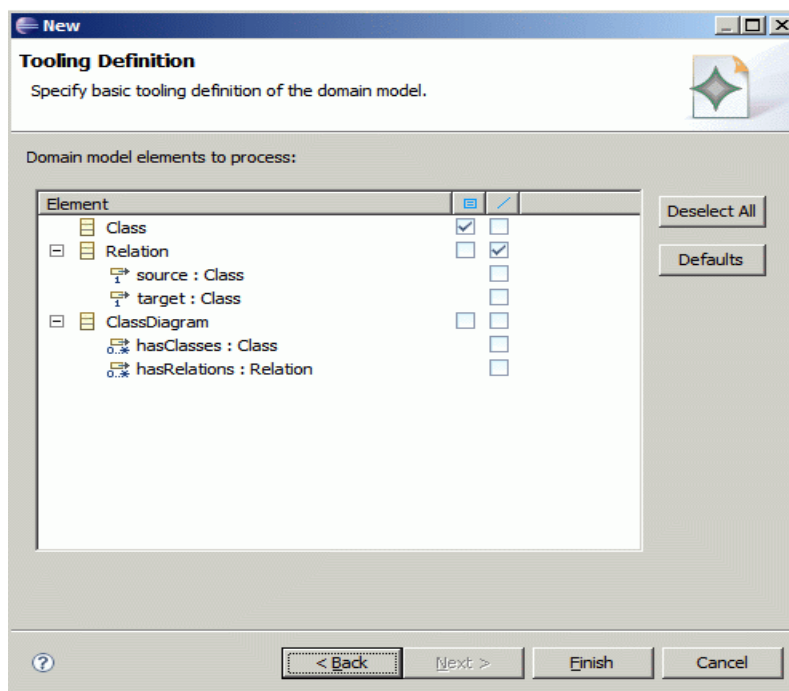
Εικ. 20. Δημιουργία Ecore μεταμοντέλου



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



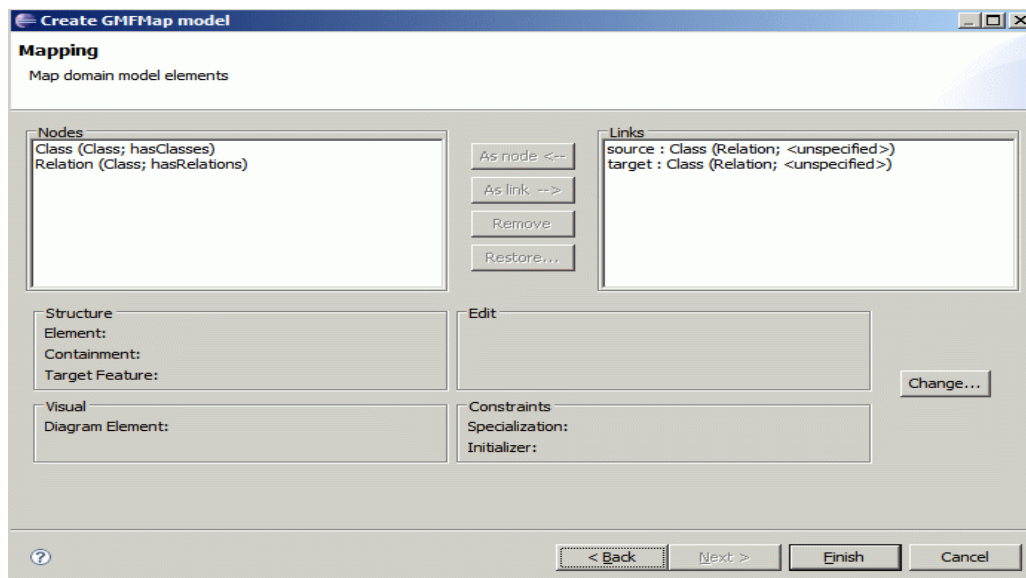
Εικ.21. Γραφικός καθορισμός με το GMF (Graphical Definition)



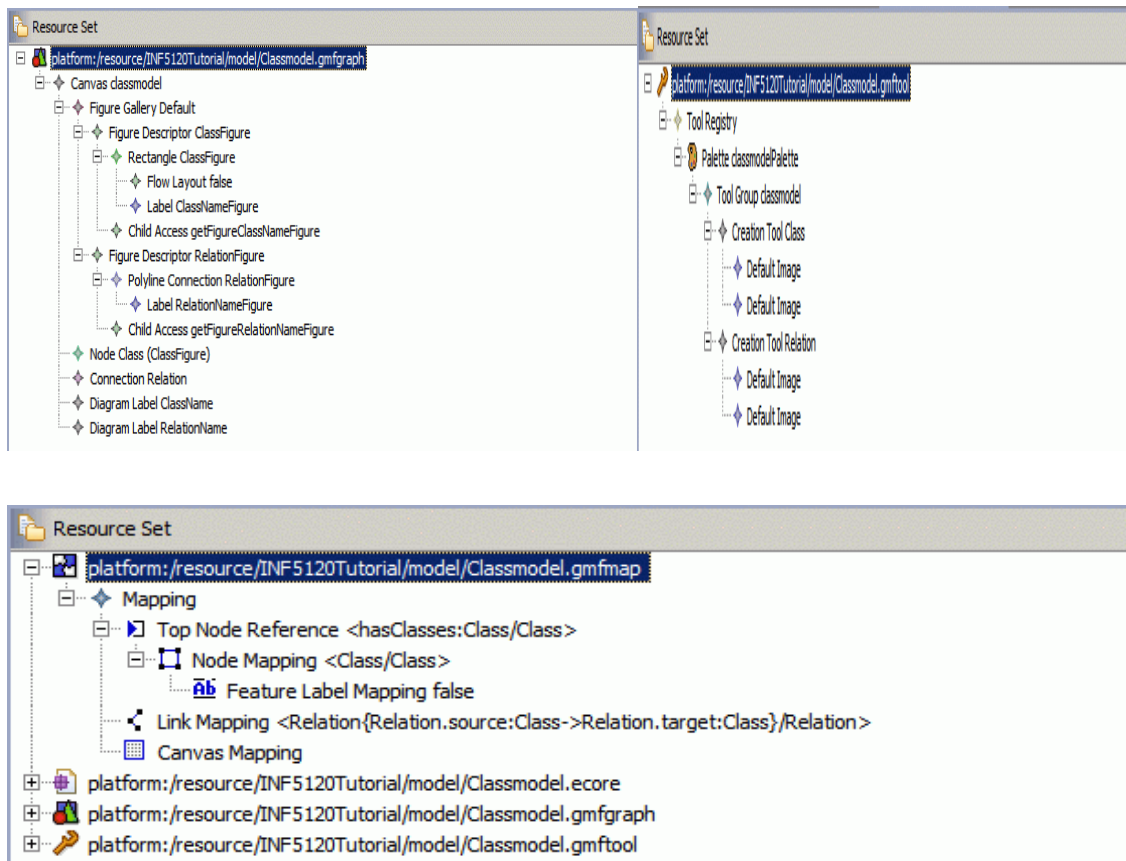
Εικ. 22. Καθορισμός σχεδίασης με το GMF (Tooling Definition)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 23. Καθορισμός χαρτογράφησης με το GMF (Mapping Definition)



Εικ. 24. Τα gmfgraph , gmftool & gmfmap αρχεία υπό δενδρική μορφή



Εικ. 25. Η επιφάνεια σχεδίασης του τελικά δημιουργούμενα γραφικού συντάκτη

2.4.2 Eclipse Epsilon - EuGENia

Η *EuGENia* είναι ένα εργαλείο το οποίο παράγει αυτόματα τα μοντέλα *.gmfgraph* , *.gmftool* και *.gmfmap* που χρειάζονται για να υλοποιήσουμε έναν GMF συντάκτη από ένα απλό “ σχολιασμένο ” (annotated) *ecore* μεταμοντέλο, επιταχύνει δε σημαντικά την πορεία υλοποίησης ενός τέτοιου συντάκτη. Για παράδειγμα , από ένα EMF μεταμοντέλο (εκφρασμένο χρησιμοποιώντας τον *Emfatic*) , μπορεί να παράγει ένα λειτουργικό GMF συντάκτη , όπως φαίνεται στην παρακάτω εικ. 26. Μπορεί να παράγει τον συντάκτη χωρίς να χρειασθεί ο χρήστης να αγγίξει κανένα από τα μοντέλα *.gmfgraph* , *.gmftool* ή *gmfmap* με το χέρι. Βέβαια το *EuGENia* δεν υποστηρίζει όλες τις επιλογές των *.gmfgraph* και *gmfmap* μοντέλων αλλά μόνο ένα μικρό υποσύνολο των συνήθως απαιτούμενων χαρακτηριστικών.

Το *EuGENia* είναι μέρος της κύριας διανομής του *Epsilon*. Η ιστοσελίδα ενημέρωσης του *Epsilon* είναι <http://download.eclipse.org/modeling/gmt/epsilon/updates> . Αν και δεν είναι απαραίτητο , θα μπορούμε να εγκαταστήσουμε το εργαλείο *Emfatic* από την ιστοσελίδα <http://www.scharf.gr/eclipse/emfatic/update/> [9] .



Όσον αφορά το *Epsilon*, είναι ένα ανοικτού κώδικα ελεύθερο λογισμικό που αποτελεί την οικογένεια γλωσσών διαχείρισης ολοκληρωμένων μοντέλων για διεργασίες όπως η πλοήγηση μοντέλων , η τροποποίηση , ο μετασχηματισμός , η επικύρωση , η σύγκριση , η συγχώνευση κ.λ.π. Μπορεί να χρησιμοποιηθεί για να διαχειρισθεί μοντέλα ή διαφορετικές τεχνολογίες μοντελοποίησης (EMF,MDR) και υποστηρίζεται από plug-ins του Eclipse (editors , outline viewers , launch configurations κλπ)

```
@namespace(uri="filesystem", prefix="filesystem")
@gmf(foo="bar")
package filesystem;

@gmf diagram(foo="bar")
class Filesystem {
    val Drive[*] drives;
    val Sync[*] syncs;
}

class Drive extends Folder {
}

class Folder extends File {
    @gmf compartment(foo="bar")
    val File[*] contents;
}

class Shortcut extends File {
    @gmf link(target decoration="arrow", style="dash")
    ref File target;
}

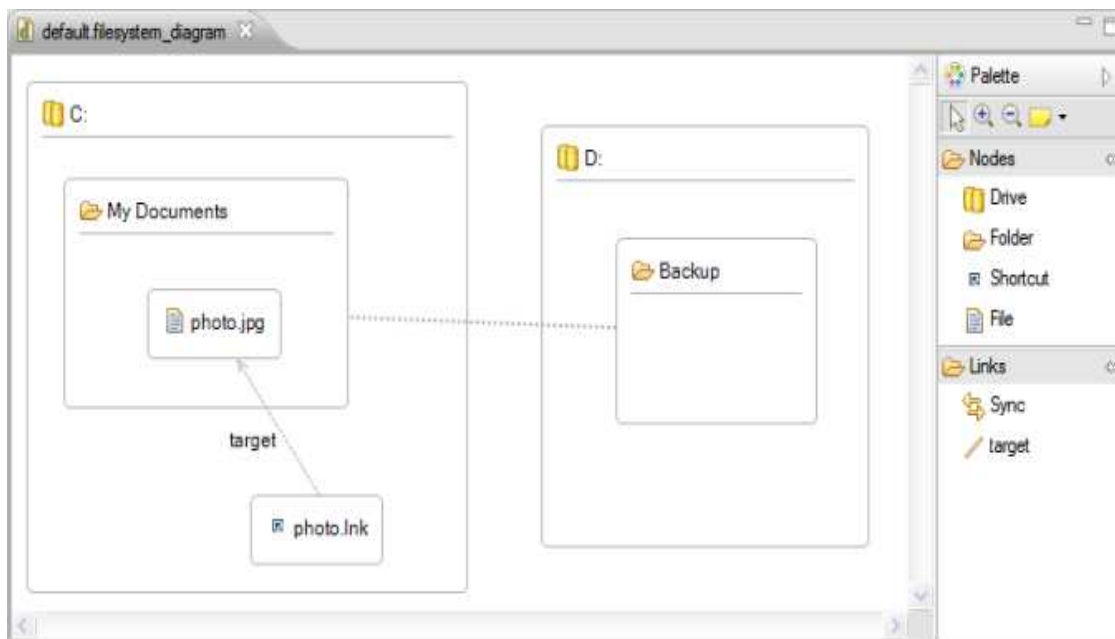
@gmf link(source="source", target="target", style="dot", width="2")
class Sync {
    ref File source;
    ref File target;
}

@gmf node(label = "name")
class File {
    attr String name;
}
```

Εικ. 26. Το 'annotated' μεταμοντέλο Filesystem



Αργυρίου Αντώνιος, Έμετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος



Εικ. 27. Παράδειγμα GMF συντάκτη που δημιουργήθηκε με το EuGENia



3. Γραφική αναπαράσταση μοντέλων ASEME

Στο Κεφάλαιο αυτό κατασκευάζονται - με γραφική αναπαράσταση - τα μεταμοντέλα της ASEME , για το παράδειγμα εργασίας «σύστημα διαχείρισης συναντήσεων» και για τις ανάγκες της παρούσας εργασίας. Το συγκεκριμένο παράδειγμα εργασίας έχει χρησιμοποιηθεί ευρέως στο παρελθόν για επίδειξη της χρήσης και άλλων τεχνολογιών π.χ. για τις μεθοδολογίες Prometheus και MAS-CommonKADS [15]. Μια σύντομη παρουσίαση των απαιτήσεων του συστήματος , για την υποστήριξη της διαδικασίας , *διαχείριση συναντήσεων* είναι η εξής : ο χρήστης χρειάζεται την βοήθεια στην διευθέτηση των συναντήσεών του , από ένα βοηθό που θα διαχειρίζεται τα 'δρομολόγιά' του και τις υπηρεσίες του. Την διαδικασία οργάνωσης των συναντήσεων διαχειρίζεται μια γραμματέας , στην οποία οι χρήστες υποβάλλουν τις αιτήσεις τους για δρομολόγηση μιας νέας συνάντησης ή αλλαγής στην ημερομηνία μιας ήδη υπάρχουσας. Η γραμματέας έρχεται σε επαφή με τους βοηθούς του χρήστη , οπότε αυτή χρειάζεται να διαπραγματευθεί μια ημερομηνία συνάντησης.

Στην παράγραφο § 3.1 γίνεται τροποποίηση των αρχικά λαμβανομένων μεταμοντέλων , από την επιστημονική κοινότητα και κατασκευάζεται εξ αρχής το μεταμοντέλο του μοντέλου πρωτοκόλλων αλληλεπίδρασης πρακτόρων (Agent Interaction Protocols model-AIP) , που δεν υπήρχε αρχικά , ενώ στην παράγραφο § 3.2 κατασκευάζονται τα γραφικά μοντέλα επεξεργασίας των αντίστοιχων μοντέλων – δηλ. οι γραφικοί συντάκτες, των SAG , SUC , AIP , SRM και IAC μοντέλων , με τα οποία δημιουργούμε τα παραδείγματα-διαγράμματα που είναι απαραίτητα για τους μετασχηματισμούς. Στα μεταμοντέλα SAG, SUC και AIP θα δημιουργηθεί μια νέα EClass με όνομα το όνομα του αντίστοιχου μοντέλου , ενώ αντί απλών associations θα δημιουργηθούν aggregations πάνω στην EClass αυτή για όλες τις άλλες , σε κάθε μεταμοντέλο , αφού έτσι προσδιορίζουν ότι ανήκουν σ' αυτήν δηλ. χωρίς την ύπαρξη αυτού του ανώτερου στοιχείου (top most element) στην ιεραρχία δεν θα υπήρχαν αυτές.

Προαπαιτούμενα για τις εργασίες αυτές αποτελούν :

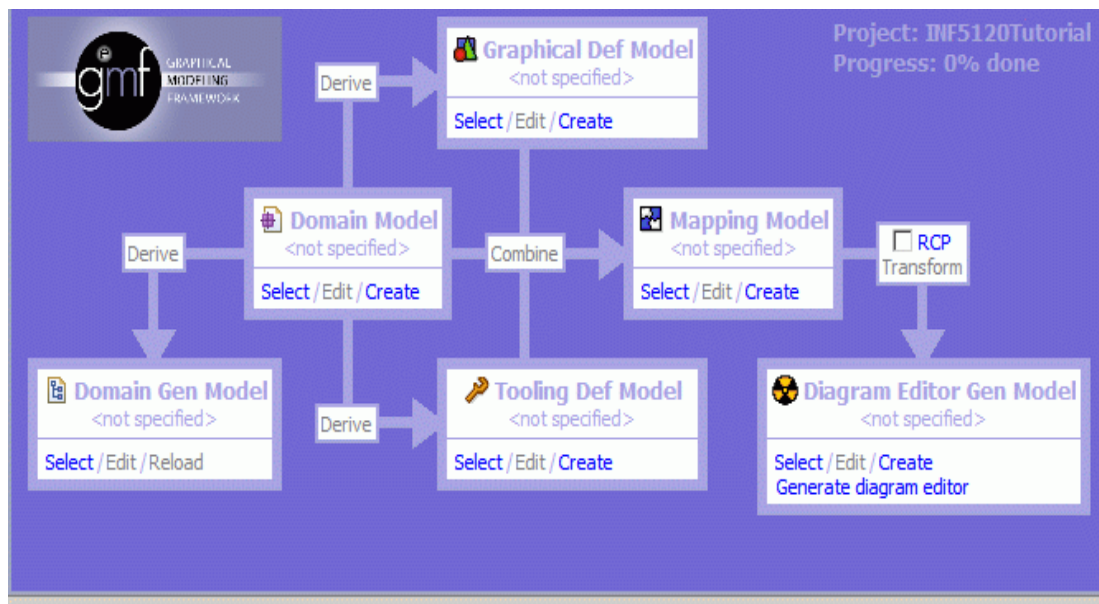
- Η εγκατάσταση της πλατφόρμας του Eclipse με τα EMF και GMF plug-ins. Εδώ χρησιμοποιήσαμε την έκδοση του Eclipse Galileo (Eclipse Project SDK,



version : 3.5.2.R35x-v20100119 , Build id :M20100211-1343) , «κατεβασμένη» από την ιστοσελίδα : <http://download.eclipse.org/eclipse/downloads>

- Βασική γνώση της χρήσης του Eclipse
- Βασική γνώση των μεταμοντέλων και της χρήσης τους και
- Βασική γνώση των γλωσσών ειδικού σκοπού (Domain Specific Languages-DSL). Για μια τέτοια γλώσσα δεν υπάρχει διεθνώς αποδεκτός καθορισμός από τί συνίσταται , αλλά προτείνεται ότι μπορούν να καθορισθούν από τέσσερα χαρακτηριστικά-κλειδιά :
 - ο Η περιοχή (domain) να είναι καλά ορισμένη και κεντρική
 - ο Η γραμματική να είναι σαφής
 - ο Η άτυπη σημασία να είναι σαφής και
 - ο Η τυπική σημασία να είναι σαφής και υλοποιήσιμη

Κατά την έναρξη δημιουργίας ενός συντάκτη διαγράμματος , μετά την εγκατάσταση του Eclipse , δυνατόν να επιλέξουμε να χρησιμοποιήσουμε το ταμπλό (dashboard) που φαίνεται στην εικ. 28 , αν και η όλη διαδικασία μπορεί να γίνει με χρήση των επιλογών της μπάρας εντολών μέσα από τον Package Explorer.



Εικ. 28. Ταμπλό του Eclipse EMF-GMF (dashboard)



Αυτό επιτυγχάνεται από την μπάρα εργαλείων μέσω των επιλογών *Window → Show View → Other...GMF Dashboard- OK*

Στην εργασία αυτή θα εργασθούμε πρωταρχικά με διαφορετικούς τύπους μοντέλων. Για να αποφύγουμε διάδοση των λαθών από μοντέλα σε μοντέλα πρέπει πάντοτε να εγκυροποιούμε (validate) το μοντέλο πριν προχωρήσουμε στο επόμενο βήμα. Αυτό γίνεται με δεξί κλικ πάνω στο μοντέλο, στον package explorer και επιλέγουμε *Validate*. Αν έχουμε προβλήματα με το μοντέλο θα εμφανισθεί μήνυμα στην *όψη προβλημάτων* δεν έχουμε ενεργό αυτήν την όψη, την προσθέτουμε μέσω των επιλογών *Window → Show View → Problems*.

3.1 Τα Νέα EMF Μεταμοντέλα

Τα μεταμοντέλα που θα αναπτυχθούν παρακάτω είναι απλές δομές που θυμίζουν διαγράμματα κλάσεων, με κλάσεις και σχέσεις μεταξύ τους για τα οποία θα δώσουμε ονόματα. Η ίδια πορεία θα ακολουθηθεί για όλα τα μεταμοντέλα, τα οποία θα παρουσιασθούν (ως .xmi αρχεία) στο παράρτημα Δ.

3.1.1 Μεταμοντέλο SAG

Με ανοικτό το Eclipse επιλέγουμε να δημιουργήσουμε ένα *άδειο emf project* : *New → Project → Empty EMF Project → SAGdesign → Finish*

Στον package explorer, στα αριστερά της οθόνης, με διπλό κλικ στο project αποκαλύπτουμε την δομή του. Δεξί κλικ στο φάκελο model και επιλέγουμε *New → Other → Ecore Model* (κάτω από το Eclipse Modeling Framework) *→ Next → name:SAG.ecore → Next → (Model Object:EPackage, XML Encoding:UTF-8) → Finish*

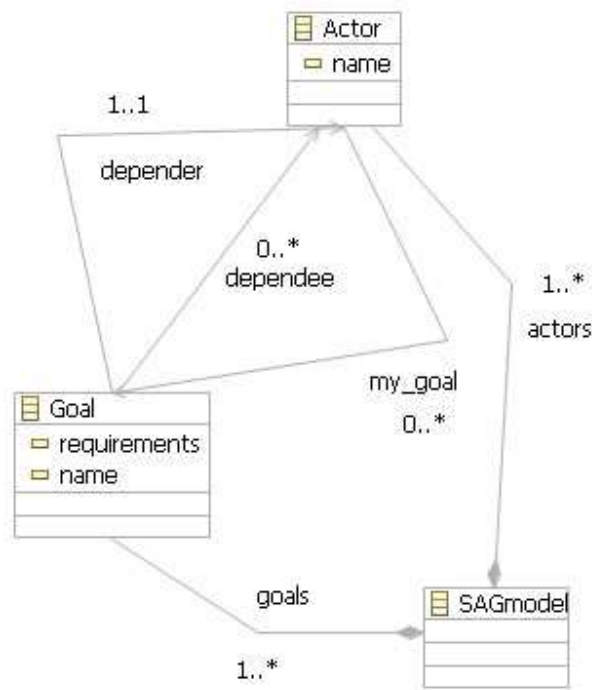
Το Model Object μας λέει ποιο είναι το ανώτερο στην ιεραρχία στοιχείο (topmost element). Όλα τα στοιχεία που θα προστεθούν στο μεταμοντέλο θα βρίσκονται κάτω από ένα πακέτο. Το επόμενο βήμα είναι να δώσουμε ένα όνομα στο ανώτερο πακέτο κάτω από το μοντέλο. Κάνουμε διπλό κλικ στο μοντέλο (ή κλικ στο σύμβολο '+') και το επεκτείνουμε, δεξί κλικ πάνω στο πακέτο null και επιλέγουμε *Show properties view →*



(EFactory Instance : EFactory , Name :SAG , NS Prefix : SAG , NS Uri : <http://www.amcl.tuc.gr/aseame/metamodels/SAG>)→Save (CTRL-S).

EFactory name είναι το όνομα το οποίο χρησιμοποιείται όταν παράγονται κάποια τμήματα κώδικα , Name είναι φυσικά το όνομα, NS Prefix είναι ένα πρόθεμα για το όνομα χώρου (namespace) και NS Uri είναι μια διεύθυνση στο διαδίκτυο για το namespace , αν υπάρχει κάπου δημοσιευμένο.Ακολουθώ , δεξί κλικ στο SAG.ecore και επιλέγουμε initialize.ecore diagram→(root element : SAGmodel)→Finish.

Στο σημείο αυτό εμφανίζεται άδεια η επιφάνεια σχεδίασης(canvas) και δημιουργούμε το μεταμοντέλο που εμφανίζεται στην εικ.29.Ρυθμίζουμε τις τιμές των ιδιότητες –μέσω της επιλογής *Show properties view* - για κάθε στοιχείο του μοντέλου , προσέχοντας τις πολλαπλότητες (για να θέσουμε πολλαπλότητα για τις aggregations πρέπει να γράψουμε 0..1 , ο συντάκτης δεν δέχεται το 0..*).Στην εικ.30 φαίνεται επίσης το μεταμοντέλο στην εκτεταμένη του μορφή.



Εικ. 29. Το νέο SAG μεταμοντέλο (ecore_diagram)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 30. Το νέο SAG μεταμοντέλο στην εκτεταμένη του μορφή

Στην συνέχεια δημιουργούμε το Domain Gen Model , ένα EMF μοντέλο που παίρνει το παραπάνω δημιουργημένο μεταμοντέλο(σε.ecore) ως είσοδο και δημιουργεί ένα μοντέλο γεννήτορα (Generator Model) που χρησιμεύει σαν βάση όταν παράγεται κώδικας που αλληλεπιδρά με το μεταμοντέλο.Πρός τούτο μπορούμε είτε με κλικ στο *derive* του GMF ταμπλό-dashboard (στον σύνδεσμο μεταξύ Domain Model και Domain Gen Model) ή με δεξί κλικ στον φάκελο model του project επιλέγουμε *New* ➔ *Other* ➔ ... ➔ *Eclipse Modeling Framework* ➔ *EMF Generator Model* ➔ *name : SAG.genmodel* ➔ ... ➔ *Finish* (βλ. εικ.31)

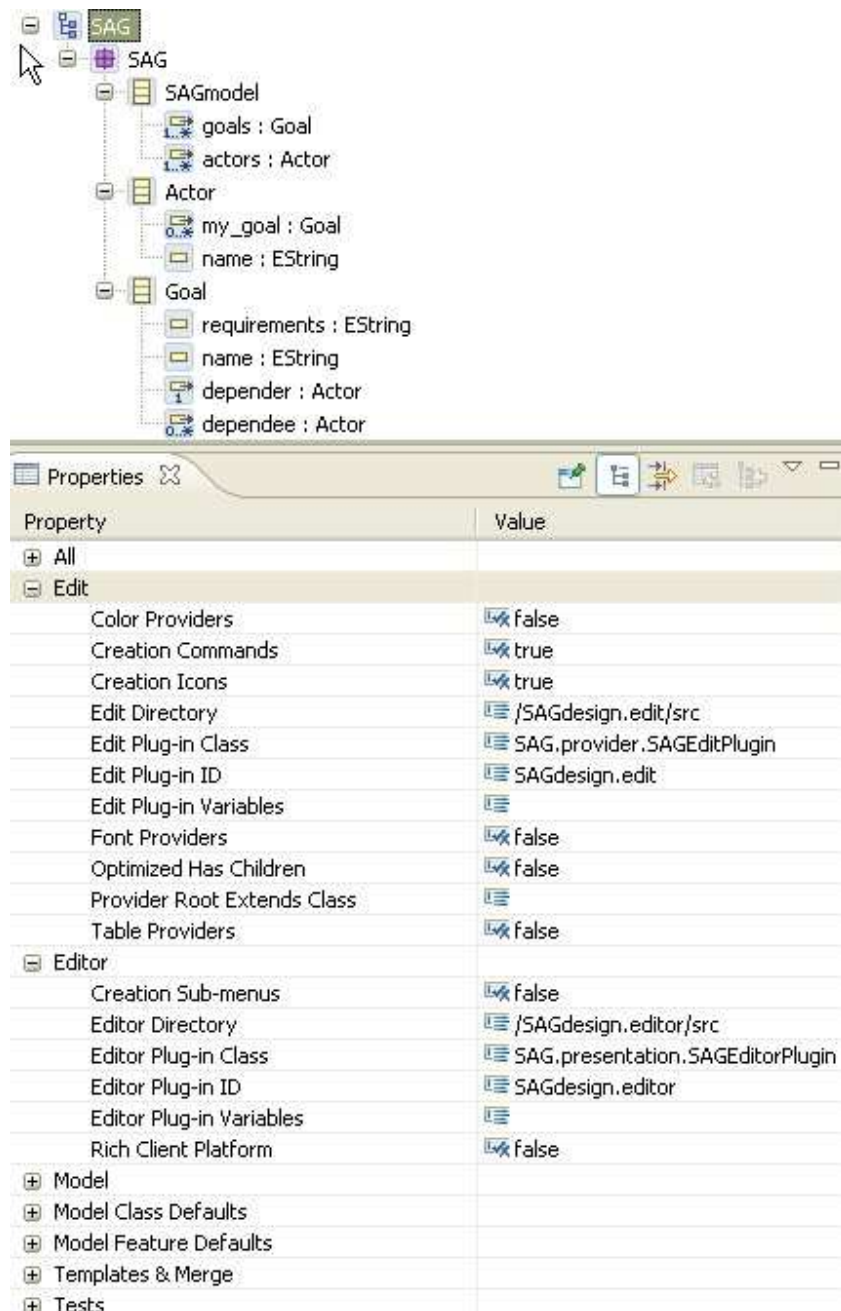
Για να παράγουμε τον κώδικα , ανοίγουμε το SAG.genmodel αρχείο , δεξί κλικ στο ανώτερο ιεραρχικά στοιχείο (top element) και επιλέγουμε *generate all*.Αυτό θα δημιουργήσει 3 νέα projects τα SAGdesign.edit , SAGdesign.editor και SAGdesign.tests , αντίστοιχα.Ο φάκελος *src* , στο EMF project που δημιουργήθηκε πρώτα θα περιέχει τον κώδικα.

3.1.2 Μεταμοντέλο SUC

Με ανοικτό το Eclipse επιλέγουμε να δημιουργήσουμε ένα άδειο *emf project* : *New* ➔ *Project* ➔ *Empty EMF Project* ➔ *name:SUCdesign* ➔ *Finish*



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 31. Το νέο SAG Generator model και η όψη των ιδιοτήτων του

Στον package explorer , στα αριστερά της οθόνης , με διπλό κλικ στο project αποκαλύπτουμε την δομή του. Δεξί κλικ στο φάκελο model και επιλέγουμε *New ➔ Other ➔ Ecore Model* (κάτω από το Eclipse Modeling Framework)

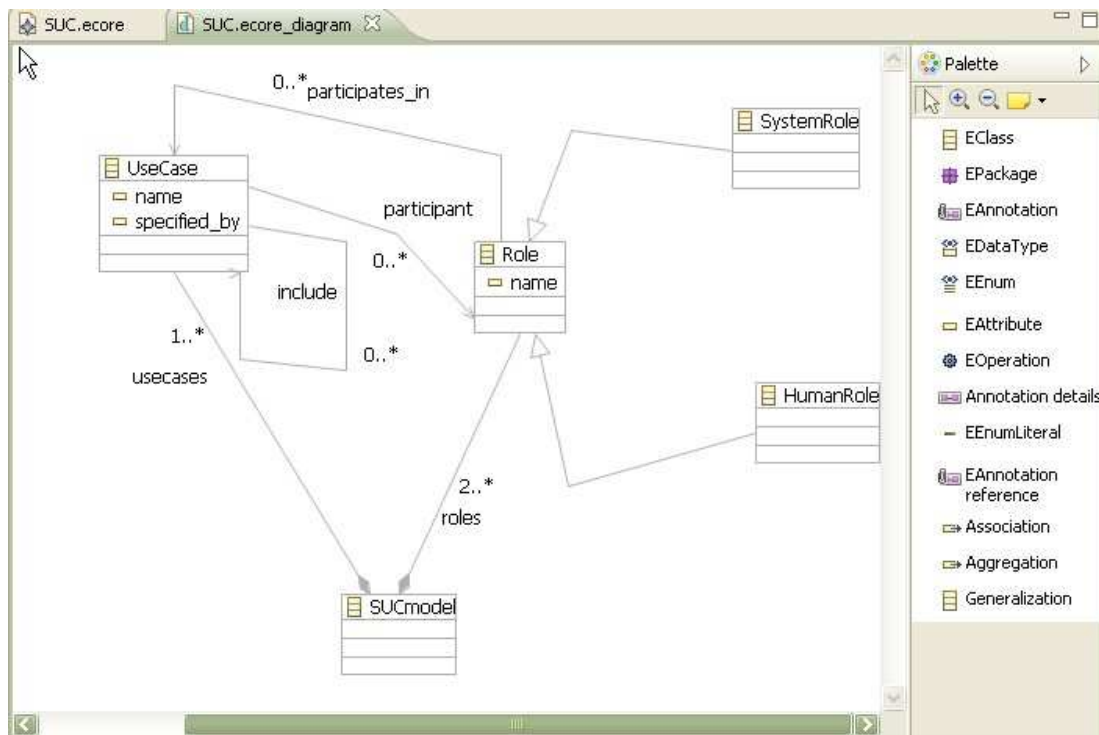


➔Next➔name:SUC.ecore➔Next➔ (Model Object:EPackage, XML Encoding:UTF-8)➔Finish

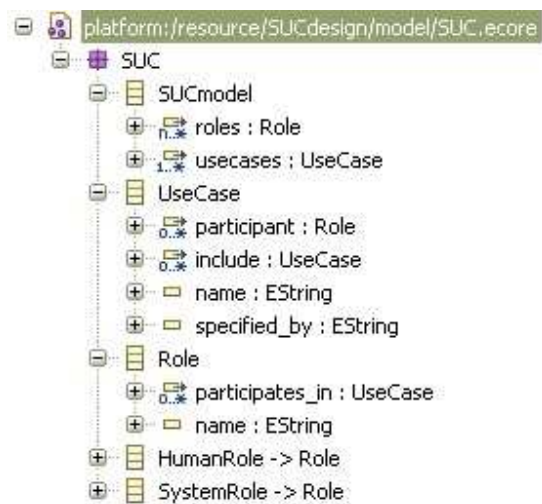
Το Model Object μας λέει ποιο είναι το ανώτερο στην ιεραρχία στοιχείο (topmost element). Όλα τα στοιχεία που θα προστεθούν στο μεταμοντέλο θα βρίσκονται κάτω από ένα πακέτο. Το επόμενο βήμα είναι να δώσουμε ένα όνομα στο ανώτερο πακέτο κάτω από το μοντέλο. Κάνουμε διπλό κλικ στο μοντέλο (ή κλικ στο σύμβολο '+') και το επεκτείνουμε, δεξί κλικ πάνω στο πακέτο null και επιλέγουμε Show properties view ➔ (EFactory Instance : EFactory, Name :SUC, NS Prefix : SUC, NS Uri : <http://www.amcl.tuc.gr/aseme/metamodels/SUC>) ➔Save (CTRL-S)

EFactory name είναι το όνομα το οποίο χρησιμοποιείται όταν παράγονται κάποια τμήματα κώδικα, Name είναι φυσικά το όνομα, NS Prefix είναι ένα πρόθεμα για το όνομα χώρου (namespace) και NS Uri είναι μια διεύθυνση στο διαδίκτυο για το namespace, αν υπάρχει κάπου δημοσιευμένο. Ακολουθώντας, δεξί κλικ στο SAG.ecore και επιλέγουμε initialize.ecore diagram ➔(root element : SUCmodel) ➔Finish.

Στο σημείο αυτό εμφανίζεται άδεια η επιφάνεια σχεδίασης (canvas) και δημιουργούμε το μεταμοντέλο που εμφανίζεται στην εικ. 32. Ρυθμίζουμε τις τιμές των ιδιοτήτων –μέσω της επιλογής Show properties view - για κάθε στοιχείο του μοντέλου, προσέχοντας τις πολλαπλότητες (για να θέσουμε πολλαπλότητα για τις aggregations πρέπει να γράψουμε 0..-1, ο συντάκτης δεν δέχεται το 0..*). Στην εικ. 33 φαίνεται επίσης το μεταμοντέλο στην εκτεταμένη του μορφή.



Εικ. 32. Το νέο SUC μεταμοντέλο (ecore_diagram)



Εικ. 33. Το νέο SUC μεταμοντέλο στην εκτεταμένη του μορφή

Στην συνέχεια δημιουργούμε το Domain Gen Model , ένα EMF μοντέλο που παίρνει το παραπάνω δημιουργημένο μεταμοντέλο(σε ecore) ως είσοδο και δημιουργεί ένα μοντέλο γεννήτορα (Generator Model) που χρησιμεύει σαν βάση όταν παράγεται



κώδικας που αλληλεπιδρά με το μεταμοντέλο.Πρός τούτο μπορούμε είτε με κλικ στο *derive* του GMF ταμπλό (στον σύνδεσμο μεταξύ Domain Model και Domain Gen Model) ή με δεξί κλικ στον φάκελο model του project επιλέγουμε *New* → *Other* → ... → *Eclipse Modeling Framework* → *EMF Generator Model* → *name* : *SUC.genmodel* → ... → *Finish*

Για να παράγουμε τον κώδικα , ανοίγουμε το *SUC.genmodel* αρχείο ,δεξί κλικ στο ανώτερο ιεραρχικά στοιχείο (top element) και επιλέγουμε *generate all*.Αυτό θα δημιουργήσει 3 νέα projects τα *SUCdesign.edit* , *SUCdesign.editor* και *SUCdesign.tests* , αντίστοιχα.Ο φάκελος *src* , στο EMF project που δημιουργήθηκε πρώτα θα περιέχει τον κώδικα.

3.1.3 Μεταμοντέλο AIP

Με ανοικτό το Eclipse επιλέγουμε να δημιουργήσουμε ένα άδειο *emf project* : *New* → *Project* → *Empty EMF Project* → *name:AIPdesign* → *Finish*

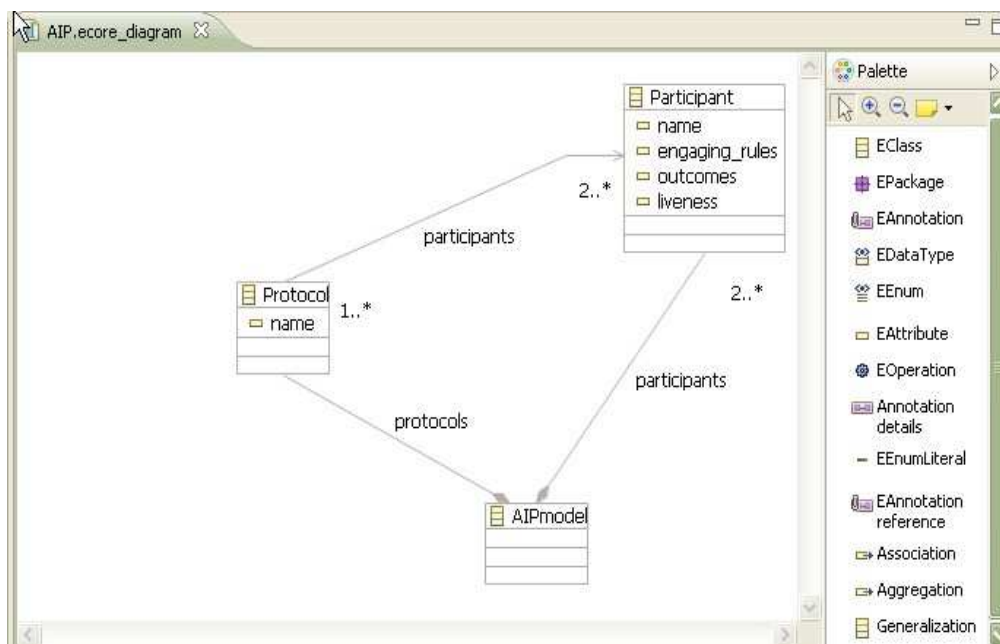
Στον package explorer , στα αριστερά της οθόνης , με διπλό κλικ στο project αποκαλύπτουμε την δομή του.Δεξί κλικ στο φάκελο model και επιλέγουμε *New* → *Other* → *Ecore Model* (κάτω από το Eclipse Modeling Framework) → *Next* → *name:AIP.ecore* → *Next* → (Model Object:EPackage, XML Encoding:UTF-8) → *Finish*

Το Model Object μας λέει ποιο είναι το ανώτερο στην ιεραρχία στοιχείο (topmost element).Όλα τα στοιχεία που θα προστεθούν στο μεταμοντέλο θα βρίσκονται κάτω από ένα πακέτο.Το επόμενο βήμα είναι να δώσουμε ένα όνομα στο ανώτερο πακέτο κάτω από το μοντέλο.Κάνουμε διπλό κλικ στο μοντέλο (ή κλικ στο σύμβολο '+') και το επεκτείνουμε , δεξί κλικ πάνω στο πακέτο null και επιλέγουμε *Show properties view* → (EFactory Instance : EFactory , Name :AIP , NS Prefix : AIP , NS Uri : <http://www.amcl.tuc.gr/aseme/metamodels/AIP>) → *Save (CTRL-S)*

EFactory name είναι το όνομα το οποίο χρησιμοποιείται όταν παράγονται κάποια τμήματα κώδικα , Name είναι φυσικά το όνομα, NS Prefix είναι ένα πρόθεμα για το όνομα χώρου (namespace) και NS Uri είναι μια διεύθυνση στο διαδίκτυο για το namespace , αν υπάρχει κάπου δημοσιευμένο.Ακολουθώς , δεξί κλικ στο *SAG.ecore* και επιλέγουμε *initialize.ecore diagram* → (root element : AIPmodel) → *Finish*.



Στο σημείο αυτό εμφανίζεται άδεια η επιφάνεια σχεδίασης(canvas) και δημιουργούμε το μεταμοντέλο που εμφανίζεται στην εικ. 34.Ρυθμίζουμε τις τιμές των ιδιότητες – μέσω της επιλογής *Show properties view* - για κάθε στοιχείο του μοντέλου , προσέχοντας τις πολλαπλότητες (για να θέσουμε πολλαπλότητα για τις aggregations πρέπει να γράψουμε 0..1 , ο συντάκτης δεν δέχεται το 0..*). Στην εικ. 35 φαίνεται επίσης το μεταμοντέλο στην εκτεταμένη του μορφή.



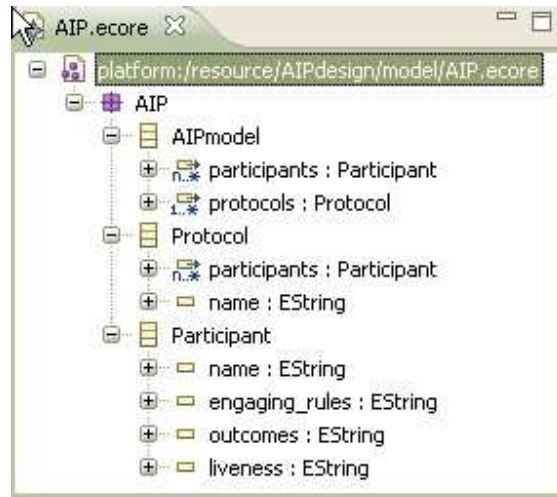
Εικ. 34. Το νέο AIP μεταμοντέλο (ecore_diagram)

Στην συνέχεια δημιουργούμε το Domain Gen Model , ένα EMF μοντέλο που παίρνει το παραπάνω δημιουργημένο μεταμοντέλο(σε ecore) ως είσοδο και δημιουργεί ένα μοντέλο γεννήτορα (Generator Model) που χρησιμεύει σαν βάση όταν παράγεται κώδικας που αλληλεπιδρά με το μεταμοντέλο.Πρός τούτο μπορούμε είτε με κλικ στο *derive* του GMF ταμπλό (στον σύνδεσμο μεταξύ Domain Model και Domain Gen Model) ή με δεξί κλικ στον φάκελο model του project επιλέγουμε *New* ➔ *Other* ➔ ... ➔ *Eclipse Modeling Framework* ➔ *EMF Generator Model* ➔ *name* : *AIP.genmodel* ➔ ... ➔

Για να παράγουμε τον κώδικα , ανοίγουμε το AIP.genmodel αρχείο ,δεξί κλικ στο ανώτερο ιεραρχικά στοιχείο (top element) και επιλέγουμε *generate all*.Αυτό θα δημιουργήσει 3 νέα projects τα AIPdesign.edit , AIPdesign.editor και AIPdesign.tests ,



αντίστοιχα.Ο φάκελος *src* , στο EMF project που δημιουργήθηκε πρώτα θα περιέχει τον κώδικα.



Εικ. 35. Το νέο AIP μεταμοντέλο στην εκτεταμένη του μορφή

3.1.4 Μεταμοντέλο SRM

Με ανοικτό το Eclipse επιλέγουμε να δημιουργήσουμε ένα άδειο *emf project* :
New → *Project* → *Empty EMF Project* → *name:SRMdesign* → *Finish*

Στον *package explorer* , στα αριστερά της οθόνης , με διπλό κλικ στο project αποκαλύπτουμε την δομή του.Δεξί κλικ στο φάκελο *model* και επιλέγουμε *New* → *Other* → *Ecore Model* (κάτω από το *Eclipse Modeling Framework*) → *Next* → *name:SRM.ecore* → *Next* → (*Model Object:EPackage, XML Encoding:UTF-8*) → *Finish*

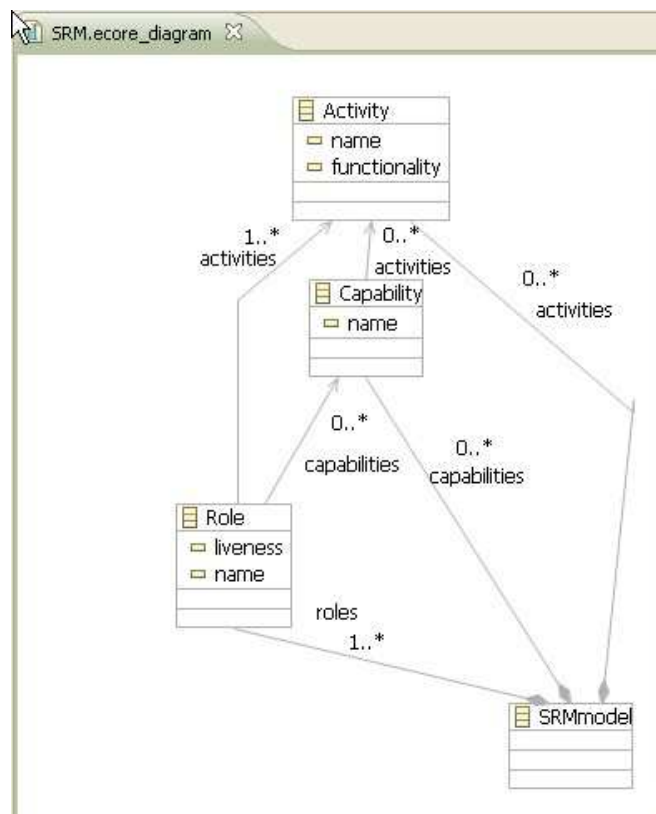
Το *Model Object* μας λέει ποιο είναι το ανώτερο στην ιεραρχία στοιχείο (*topmost element*).Όλα τα στοιχεία που θα προστεθούν στο μεταμοντέλο θα βρίσκονται κάτω από ένα πακέτο.Το επόμενο βήμα είναι να δώσουμε ένα όνομα στο ανώτερο πακέτο κάτω από το μοντέλο.Κάνουμε διπλό κλικ στο μοντέλο (ή κλικ στο σύμβολο '+') και το επεκτείνουμε , δεξί κλικ πάνω στο πακέτο *null* και επιλέγουμε *Show properties view* → (*EFactory Instance : EFactory , Name :SRM , NS Prefix : SRM , NS Uri : http://www.amcl.tuc.gr/aseme/metamodels/SRM*) → *Save (CTRL-S)*

EFactory name είναι το όνομα το οποίο χρησιμοποιείται όταν παράγονται κάποια τμήματα κώδικα , *Name* είναι φυσικά το όνομα, *NS Prefix* είναι ένα πρόθεμα για το



όνομα χώρου (namespace) και NS Uri είναι μια διεύθυνση στο διαδίκτυο για το namespace , αν υπάρχει κάπου δημοσιευμένο.Ακολούθως , δεξί κλικ στο SAG.ecore και επιλέγουμε initialize.ecore diagram→(root element : SRMmodel)→Finish.

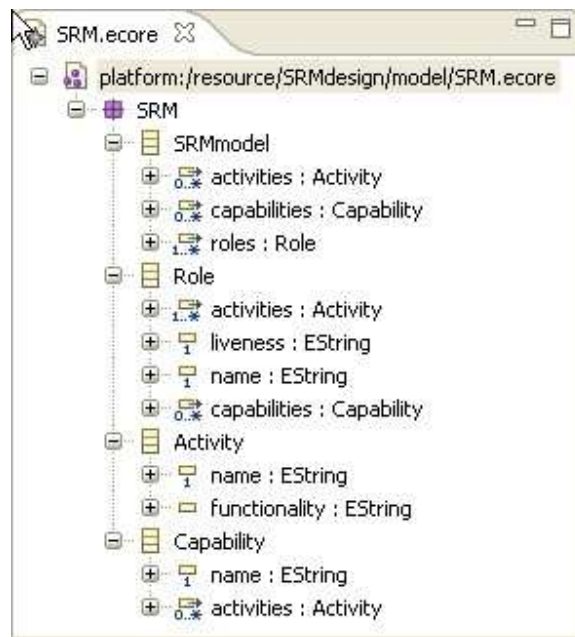
Στο σημείο αυτό εμφανίζεται άδεια η επιφάνεια σχεδίασης(canvas) και δημιουργούμε το μεταμοντέλο που εμφανίζεται στην εικ. 36.Ρυθμίζουμε τις τιμές των ιδιοτήτες –μέσω της επιλογής *Show properties view* - για κάθε στοιχείο του μοντέλου , προσέχοντας τις πολλαπλότητες (για να θέσουμε πολλαπλότητα για τις aggregations πρέπει να γράψουμε 0..1 , ο συντάκτης δεν δέχεται το 0..*). Στην εικ. 37 φαίνεται επίσης το μεταμοντέλο στην εκτεταμένη του μορφή.



Εικ. 36. Το νέο SRM μεταμοντέλο (.ecore_diagram)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 37. Το νέο SRM μεταμοντέλο στην εκτεταμένη του μορφή

Στην συνέχεια δημιουργούμε το Domain Gen Model , ένα EMF μοντέλο που παίρνει το παραπάνω δημιουργημένο μεταμοντέλο(σε.ecore) ως είσοδο και δημιουργεί ένα μοντέλο γεννήτορα (Generator Model) που χρησιμεύει σαν βάση όταν παράγεται κώδικας που αλληλεπιδρά με το μεταμοντέλο.Πρός τούτο μπορούμε είτε με κλικ στο *derive* του GMF ταμπλό (στον σύνδεσμο μεταξύ Domain Model και Domain Gen Model) ή με δεξί κλικ στον φάκελο model του project επιλέγουμε *New* ➔ *Other* ➔ ... ➔ *Eclipse Modeling Framework* ➔ *EMF Generator Model* ➔ *name : SRM.genmodel* ➔ ... ➔ *Finish*

Για να παράγουμε τον κώδικα , ανοίγουμε το SRM.genmodel αρχείο ,δεξί κλικ στο ανώτερο ιεραρχικά στοιχείο (top element) και επιλέγουμε *generate all*.Αυτό θα δημιουργήσει 3 νέα projects τα SRMdesign.edit , SRMdesign.editor και SRMdesign.tests , αντίστοιχα.Ο φάκελος *src* , στο EMF project που δημιουργήθηκε πρώτα θα περιέχει τον κώδικα.



3.1.5 Μεταμοντέλο IAC

Με ανοικτό το Eclipse επιλέγουμε να δημιουργήσουμε ένα άδειο *emf project* :
New → *Project* → *Empty EMF Project* → *name:IACdesign* → *Finish*

Στον *package explorer* , στα αριστερά της οθόνης , με διπλό κλικ στο *project* αποκαλύπτουμε την δομή του. Δεξί κλικ στο φάκελο *model* και επιλέγουμε *New* → *Other* → *Ecore Model* (κάτω από το *Eclipse Modeling Framework*) → *Next* → *name:IAC.ecore* → *Next* → (*Model Object:EPackage, XML Encoding:UTF-8*) → *Finish*

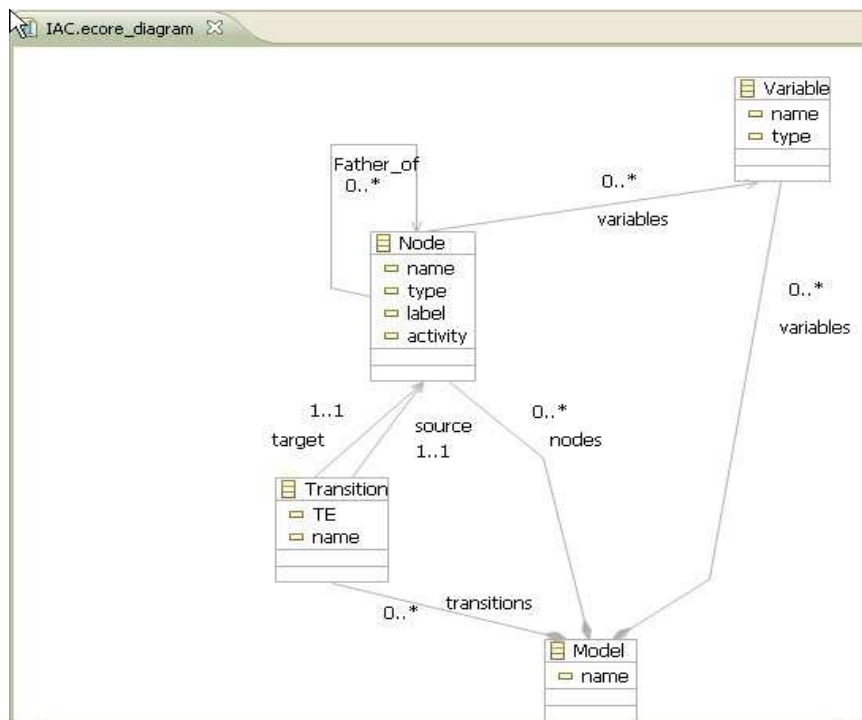
Το *Model Object* μας λέει ποιο είναι το ανώτερο στην ιεραρχία στοιχείο (*topmost element*). Όλα τα στοιχεία που θα προστεθούν στο μεταμοντέλο θα βρίσκονται κάτω από ένα πακέτο. Το επόμενο βήμα είναι να δώσουμε ένα όνομα στο ανώτερο πακέτο κάτω από το μοντέλο. Κάνουμε διπλό κλικ στο μοντέλο (ή κλικ στο σύμβολο '+') και το επεκτείνουμε , δεξί κλικ πάνω στο πακέτο *null* και επιλέγουμε *Show properties view* → (*EFactory Instance : EFactory , Name :IAC , NS Prefix : IAC , NS Uri : http://www.amcl.tuc.gr/aseme/metamodels/IAC*) → *Save (CTRL-S)*

EFactory name είναι το όνομα το οποίο χρησιμοποιείται όταν παράγονται κάποια τμήματα κώδικα , *Name* είναι φυσικά το όνομα, *NS Prefix* είναι ένα πρόθεμα για το όνομα χώρου (*namespace*) και *NS Uri* είναι μια διεύθυνση στο διαδίκτυο για το *namespace* , αν υπάρχει κάπου δημοσιευμένο. Ακολουθώ , δεξί κλικ στο *IAC.ecore* και επιλέγουμε *initialize.ecore diagram* → (*root element : IACmodel*) → *Finish*.

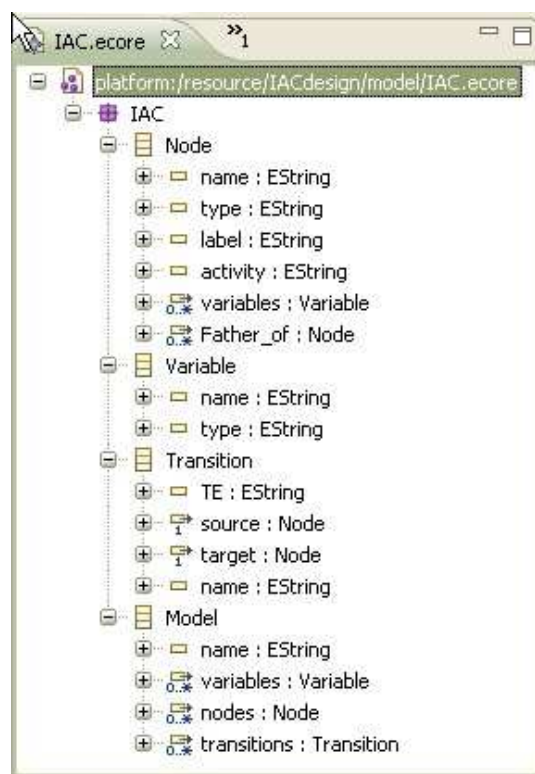
Στο σημείο αυτό εμφανίζεται άδεια η επιφάνεια σχεδίασης (*canvas*) και δημιουργούμε το μεταμοντέλο που εμφανίζεται στην εικ. 38. Ρυθμίζουμε τις τιμές των ιδιοτήτων –μέσω της επιλογής *Show properties view* - για κάθε στοιχείο του μοντέλου , προσέχοντας τις πολλαπλότητες (για να θέσουμε πολλαπλότητα για τις *aggregations* πρέπει να γράψουμε *0..-1* , ο συντάκτης δεν δέχεται το *0..**). Στην εικ. 39 φαίνεται επίσης το μεταμοντέλο στην εκτεταμένη του μορφή.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 38. Το νέο IAC μεταμοντέλο (ecore_diagram)



Εικ. 39. Το νέο IAC μεταμοντέλο στην εκτεταμένη του μορφή



Στην συνέχεια δημιουργούμε το Domain Gen Model , ένα EMF μοντέλο που παίρνει το παραπάνω δημιουργημένο μεταμοντέλο(σε.ecore) ως είσοδο και δημιουργεί ένα μοντέλο γεννήτορα (Generator Model) που χρησιμεύει σαν βάση όταν παράγεται κώδικας που αλληλεπιδρά με το μεταμοντέλο.Πρός τούτο μπορούμε είτε με κλικ στο *derive* του GMF ταμπλό (στον σύνδεσμο μεταξύ Domain Model και Domain Gen Model) ή με δεξί κλικ στον φάκελο model του project επιλέγουμε *New ➔ Other ➔ ... ➔ Eclipse Modeling Framework ➔ EMF Generator Model ➔ name : IAC.genmodel ➔ ... ➔ Finish*

Για να παράγουμε τον κώδικα , ανοίγουμε το IAC.genmodel αρχείο ,δεξί κλικ στο ανώτερο ιεραρχικά στοιχείο (top element) και επιλέγουμε *generate all*.Αυτό θα δημιουργήσει 3 νέα projects τα IACdesign.edit , IACdesign.editor και IACdesign.tests , αντίστοιχα.Ο φάκελος *src* , στο EMF project που δημιουργήθηκε πρώτα θα περιέχει τον κώδικα.

ΠΑΡΑΤΗΡΗΣΗ για τα λάθη

Στην περίπτωση που υπάρχουν λάθη στα projects που δημιουργήθηκαν , αυτό συχνά οφείλεται σε προβλήματα στο Ecore αρχείο.Ένα από αυτά μπορεί να αναφερθεί όταν παράγεται το EMF μοντέλο και αφορά κακόμορφο NS Prefix.Αν πάρουμε τέτοιο λάθος , βάζουμε νέο όνομα στο NS Prefix , αποθηκεύουμε το Ecore αρχείο και βάζουμε το αρχικό όνομα πάλι και πάλι το αποθηκεύουμε.Αυτή η διαδικασία θα πρέπει πιθανώς να διορθώσει το πρόβλημα.Αν χρειασθεί να γυρίσουμε πίσω και να κάνουμε αλλαγές στο Ecore μοντέλο διαγράφουμε τα παραγόμενα projects , τον κώδικα κάτω απ' τον φάκελο *src* και τα αρχεία MANIFEST.MF στον φάκελο META-INF , built.properties , plugin.properties και plugin.xml για να πάρουμε μια «καθαρή κατάσταση».

3.2 Τα γραφικά εργαλεία

Τέλος κατασκευάζονται τα γραφικά μοντέλα επεξεργασίας των αντίστοιχων μοντέλων δηλ. οι γραφικοί συντάκτες, των SAG , SUC , AIP , SRM και IAC μοντέλων, με τα οποία δημιουργούμε τα παραδείγματα-διαγράμματα που είναι απαραίτητα για τους μετασχηματισμούς.



Θα αναπτύξουμε 3 μοντέλα της GMF γλώσσας ειδικού σκοπού (GMF Domain Specific Language-DSL) για κάθε μεταμοντέλο. Αυτά είναι το μοντέλο γραφικού καθορισμού (Graphical Definition Model), το μοντέλο καθορισμού σχεδίασης (Tooling Definition Model) και το μοντέλο καθορισμού χαρτογράφησης (Mapping Definition Model).

Το πρώτο παρουσιάζει όλα τα γραφικά στοιχεία που θα περιληφθούν στο διάγραμμα. Το δεύτερο παρουσιάζει ότι εργαλείο διατίθεται, βασιζόμενοι σ' αυτό μπορούμε να δημιουργήσουμε εργαλεία όπως είναι η παλέτα στον Ecore συντάκτη, όπως είδαμε προηγουμένως. Το τρίτο συνδέει το μεταμοντέλο, το γραφικό μοντέλο και το μοντέλο εργαλείων μαζί. Στοιχεία γραφικά και εργαλείων συνδέονται με τα αντίστοιχα τμήματά τους στο μεταμοντέλο.

3.2.1 Γραφικό μοντέλο επεξεργασίας SAG διαγραμμάτων

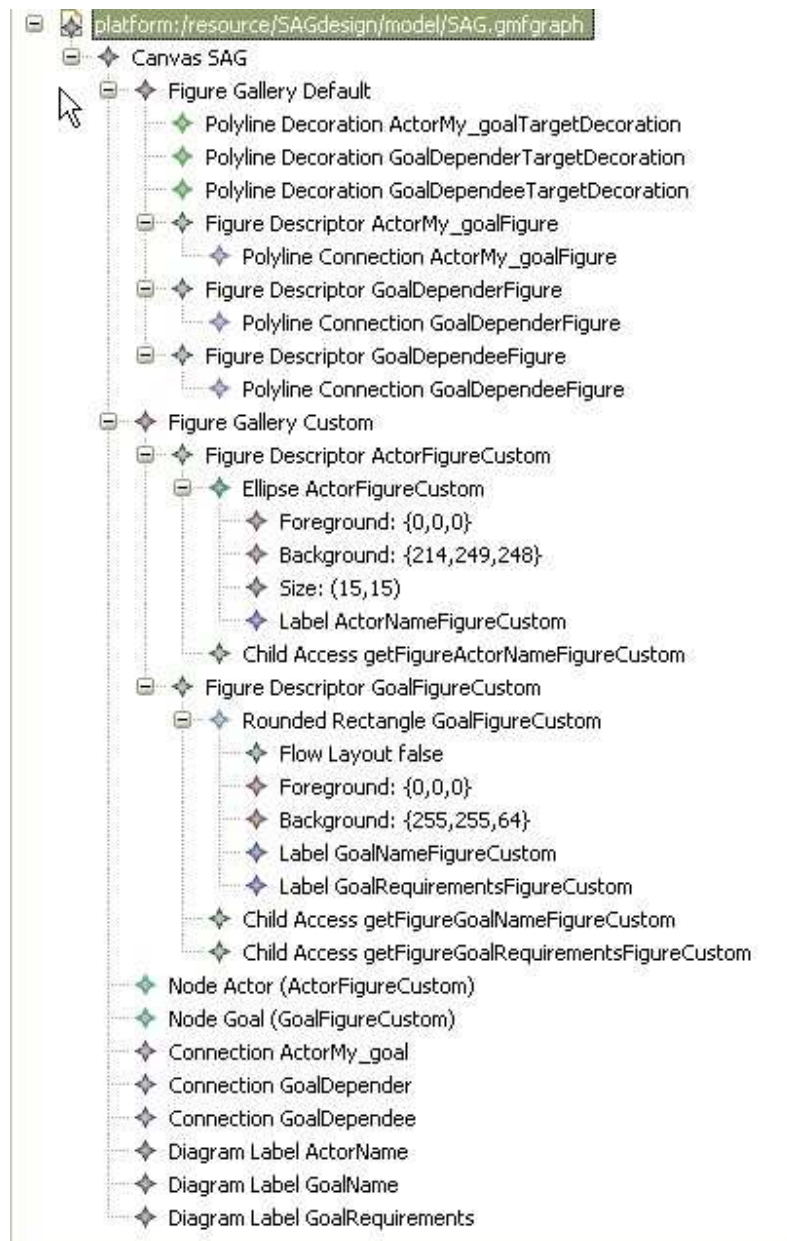
Ανάπτυξη του μοντέλου γραφικού καθορισμού (Graphical Definition Model)

Αρχίζουμε την ανάπτυξη του μοντέλου γραφικού καθορισμού είτε κλικαροντας στο *Derive* σε *Domain Model* και *Graphical Def Model* στο GMF ταμπλό (dashboard) ή απλά δεξί κλικ στον φάκελο *model* και επιλέγοντας *New* → *Other* → *Graphical Modeling Framework* → *Simple Graphical Definition Model* → *name:SAG.gmfgraph* → *Next* → *Load ecore file* → *Diagram element : SAGmodel* → επιλογή στοιχείων που θα παραχθούν στην αρχική έκδοση του γραφικού μοντέλου → *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 40.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 40. Το SAG Graphical Definition Model

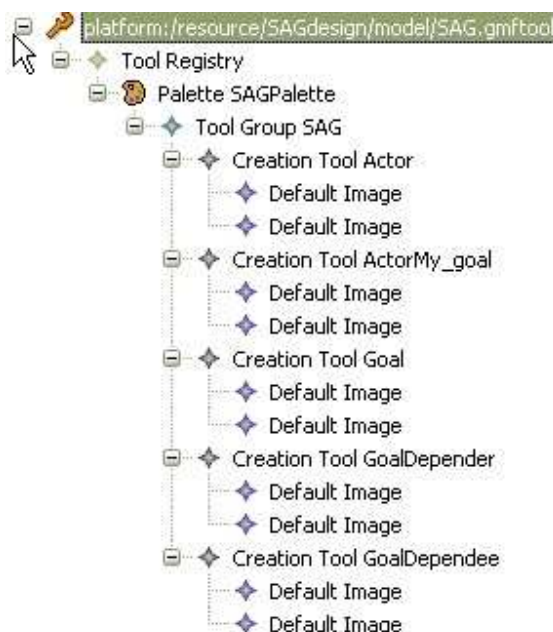
Υπάρχει μια διάκριση εδώ ως προς τις αναφορές , δηλώνοντας τί θα περιέχει η επιφάνεια σχεδίασης (Node , Connection) και πώς θα παρουσιάζονται οπτικά. Αυτό μας επιτρέπει να διαφοροποιήσουμε την απεικόνιση κάποιων στοιχείων χωρίς να διαγράφουμε και να επαναπροσθέτουμε κ.ο.κ.



Ανάπτυξη του μοντέλου καθορισμού σχεδίασης(Tooling Definition Model)

Ακολουθεί η δημιουργία των εργαλείων που θα χρησιμοποιήσουμε για να δημιουργήσουμε κλάσεις και σχέσεις στο διάγραμμα.Είτε αρχίζοντας από το GMF ταμπλό (dashboard) είτε με *New* ➔ *Other* ➔ *Graphical Modeling Framework* ➔ *Simple Tooling Definition Model* ➔ *name : SAG.gmftool* ➔ *Next* ➔ *Load Ecore file* ➔ *Next* ➔ ... ➔ επιλογή στοιχείων (κλάσεων και σχέσεων) ➔ *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 42



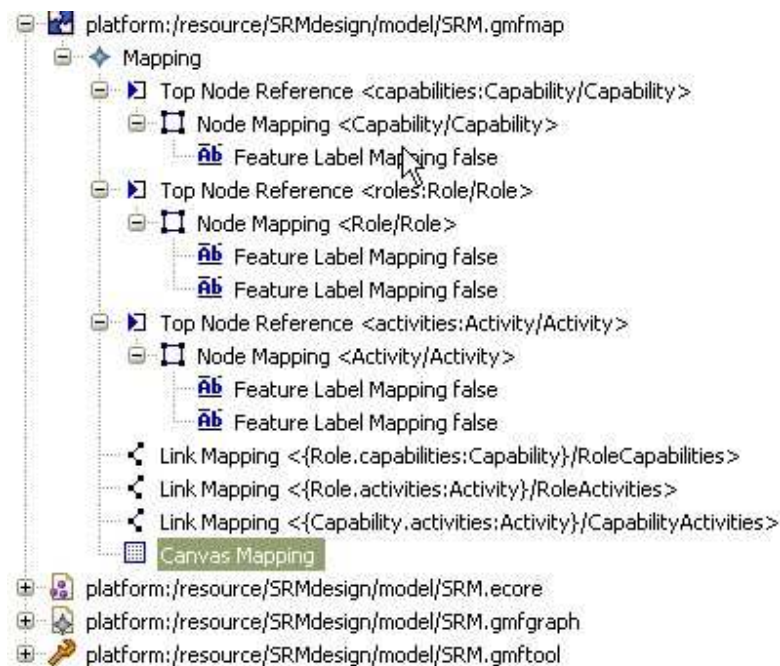
Εικ. 41. Το SAG Tooling Definition Model

Ανάπτυξη του μοντέλου καθορισμού χαρτογράφησης (Mapping Definition Model)

Ως τελικό βήμα στην δημιουργία του συγκεκριμένου συντάκτη , θα ενώσουμε το Ecore μεταμοντέλο , το γραφικό μοντέλο και το σχεδιαστικό μοντέλο μαζί σε ένα τρίτο GMF μοντέλο , το μοντέλο καθορισμού χαρτογράφησης.Αρχίζοντας όπως πάντα είτε από το GMF ταμπλό (dashboard) είτε με *New* ➔ *Other* ➔ *Graphical Modeling Framework* ➔ *Guide Mapping Model Creation* ➔ *name : SAG.gmfmap* ➔μέσω του οδηγού και κάνοντας τις κατάλληλες επιλογές θα καταλήξουμε να πάρουμε το παρακάτω αρχείο σε εκτεταμένη μορφή εικ. 42.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εκ. 42. Το SAG Mapping Definition Model

Δημιουργώντας και εκτελώντας το διάγραμμα συντάκτη (Generating & Running the Diagram editor)

Είτε πατώντας κλικ στην επιλογή *transform* (στο ταμπλό *GMF-dashboard*) μεταξύ *Mapping Model* και *Diagram Editor Gen Model* και ακολουθώντας τον οδηγό, είτε με δεξί κλικ στο *SAG.gmfmap* αρχείο → *Create Generator Model* → *name* : *SAG.gmfgen* → ... → *Finish*, παίρνουμε ένα μοντέλο γεννήτριας για το διάγραμμά μας.

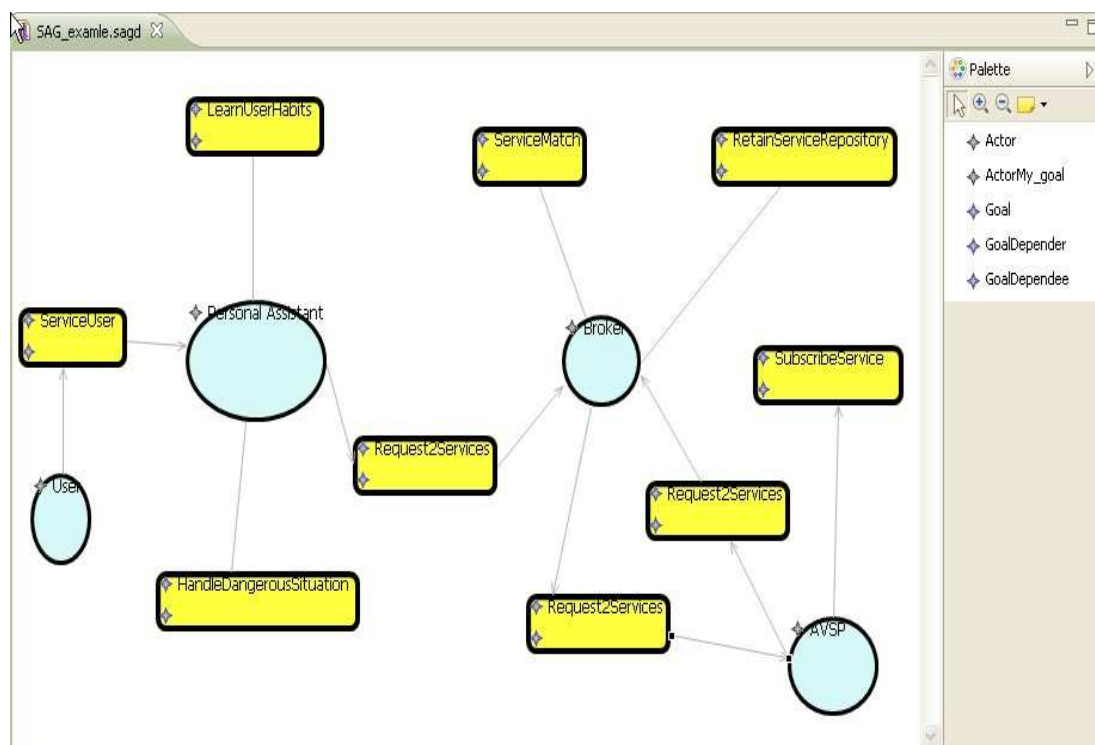
Στην συνέχεια δημιουργούμε τον κώδικα του συντάκτη διαγράμματος είτε ανοίγοντας το *gmfgen* μοντέλο και δεξί κλικ σ' αυτό → *Generate Diagram Code* ή το ίδιο στο *gmf* ταμπλό (*dashboard*). Δημιουργείται έτσι ένα νέο project στον χώρο εργασίας μας, το *SAGdesign.diagram*. Για να τρέξουμε τον συντάκτη διαγράμματος κάνουμε δεξί κλικ στο project αυτό και επιλέγουμε *Run AS-> Eclipse Application*. Για να κατασκευάσουμε ένα νέο διάγραμμα, δημιουργούμε ένα νέο project στο νέο στιγμιότυπο του *Eclipse*, επιλέγουμε *New* → *Other* και θα βρούμε το *SAG Diagram* ως επιλογή κάτω από τον



φάκελο Examples. Ακολουθώντας τον οδηγό που εμφανίζεται κατασκευάζουμε το νέο μας διάγραμμα.

Μπορούμε να κατασκευάζουμε νέα διαγράμματα χωρίς να ανοίγουμε ένα νέο στιγμιότυπο του Eclipse, αλλά μέσα από το αρχικό στιγμιότυπο. Προς τούτο, κάνουμε δεξί κλικ πάνω στο νέο plug-in (project) που δημιουργήσαμε (SAGdesign.diagram) → Export → Deployable plug-ins and fragments → επιλογή plug-ins SAGdesign – SAGdesign.diagram-SAGdesign.edit & SAGdesign.editor → επιλογή φακέλου προορισμού → Finish

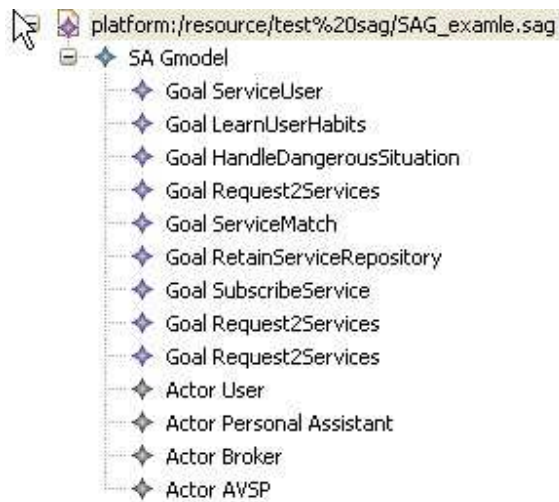
Τελικά αντιγράφουμε τα παραγόμενα .jar αρχεία στον φάκελο Eclipse plug-ins και ξανα-ανοίγουμε το Eclipse. Μπορούμε τώρα να κατασκευάσουμε μοντέλα SAG επιλέγοντας File → New → Other και ψάχνοντας για το διάγραμμα SAG. Με αυτό τον τρόπο κατασκευάσαμε τα αρχεία SAG_example.sagd & SAG_example.sag, για τις ανάγκες της εργασίας αυτής (βλ. εικ. 43 & 44)



Εικ. 43. Το νέο SAG μοντέλο (όνομα αρχείου SAG_sample.sagd)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 44. Το νέο Domain SAG μοντέλο (όνομα αρχείου SAG_sample.sag)

3.2.2 Γραφικό μοντέλο επεξεργασίας SUC διαγραμμάτων

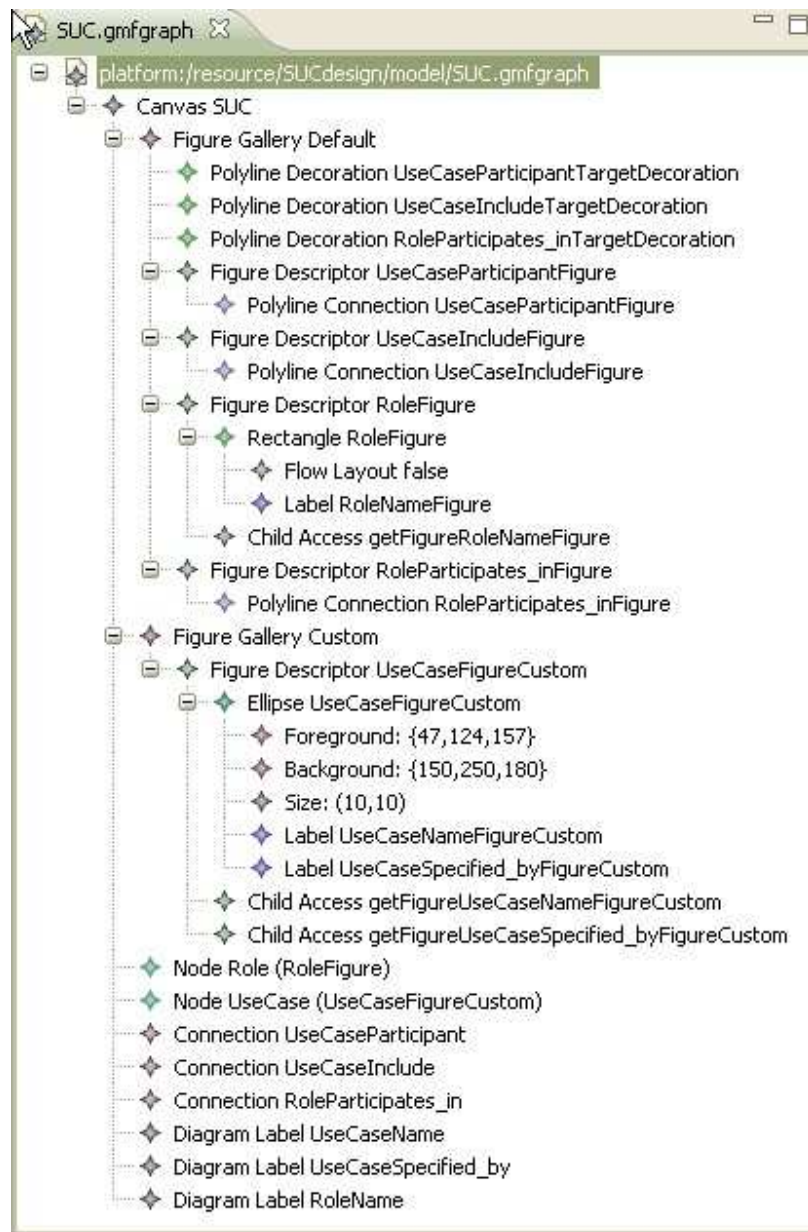
Ανάπτυξη του μοντέλου γραφικού καθορισμού (Graphical Definition Model)

Αρχίζουμε την ανάπτυξη του μοντέλου γραφικού καθορισμού είτε κλικάροντας στο *Derive* σε *Domain Model* και *Graphical Def Model* στο GMF ταμπλό (dashboard) ή απλά δεξί κλικ στον φάκελο *model* και επιλέγοντας *New* → *Other* → *Graphical Modeling Framework* → *Simple Graphical Definition Model* → *name:SUC.gmfgraph* → *Next* → *Load ecore file* → *Diagram element : SUCmodel* → επιλογή στοιχείων που θα παραχθούν στην αρχική έκδοση του γραφικού μοντέλου → *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 45.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 45. Το SUC Graphical Definition Model

Ανάπτυξη του μοντέλου καθορισμού σχεδίασης(Tooling Definition Model)

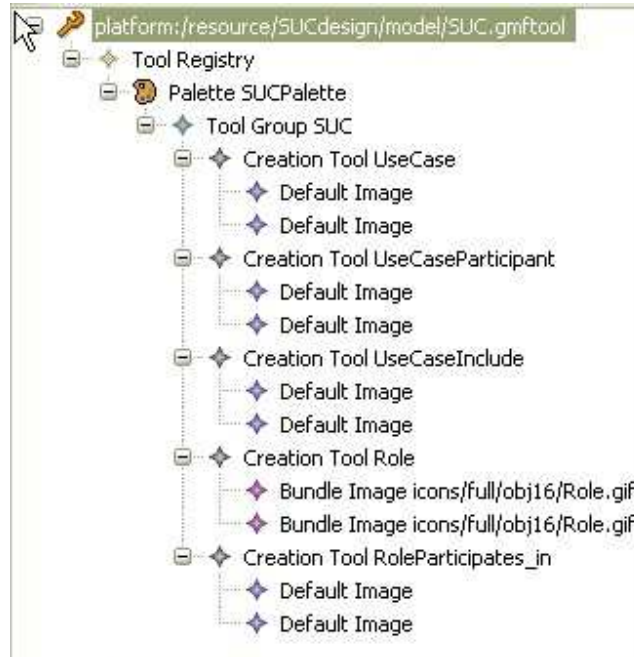
Ακολουθεί η δημιουργία των εργαλείων που θα χρησιμοποιήσουμε για να δημιουργήσουμε κλάσεις και σχέσεις στο διάγραμμα.Είτε αρχίζοντας από το GMF ταμπλό (dashboard) είτε με *New ➔ Other ➔ Graphical Modeling Framework ➔ Simple*



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'

Tooling Definition Model ➔name : SUC.gmftool ➔Next ➔Load Ecore file ➔Next ➔... ➔ επιλογή στοιχείων (κλάσεων και σχέσεων) ➔Finish

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 46.



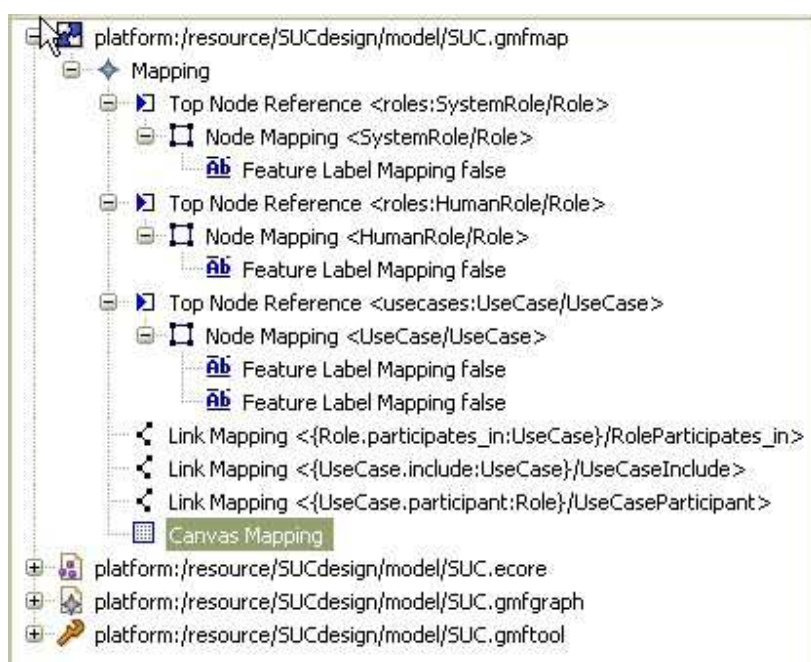
Εικ. 46. Το SUC Tooling Definition Model

Ανάπτυξη του μοντέλου καθορισμού χαρτογράφησης (Mapping Definition Model)

Ως τελικό βήμα στην δημιουργία του συγκεκριμένου συντάκτη , θα ενώσουμε το Ecore μεταμοντέλο , το γραφικό μοντέλο και το σχεδιαστικό μοντέλο μαζί σε ένα τρίτο GMF μοντέλο , το μοντέλο καθορισμού χαρτογράφησης. Αρχίζοντας όπως πάντα είτε από το GMF ταμπλό (dashboard) είτε με *New ➔Other ➔Graphical Modeling Framework ➔Guide Mapping Model Creation ➔name : SUC.gmfmap ➔.....* μέσω του οδηγού και κάνοντας τις κατάλληλες επιλογές θα καταλήξουμε να πάρουμε το παρακάτω αρχείο σε εκτεταμένη μορφή εικ. 47.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εκ. 47. Το SUC Mapping Definition Model

Δημιουργώντας και εκτελώντας το διάγραμμα συντάκτη (Generating & Running the Diagram editor)

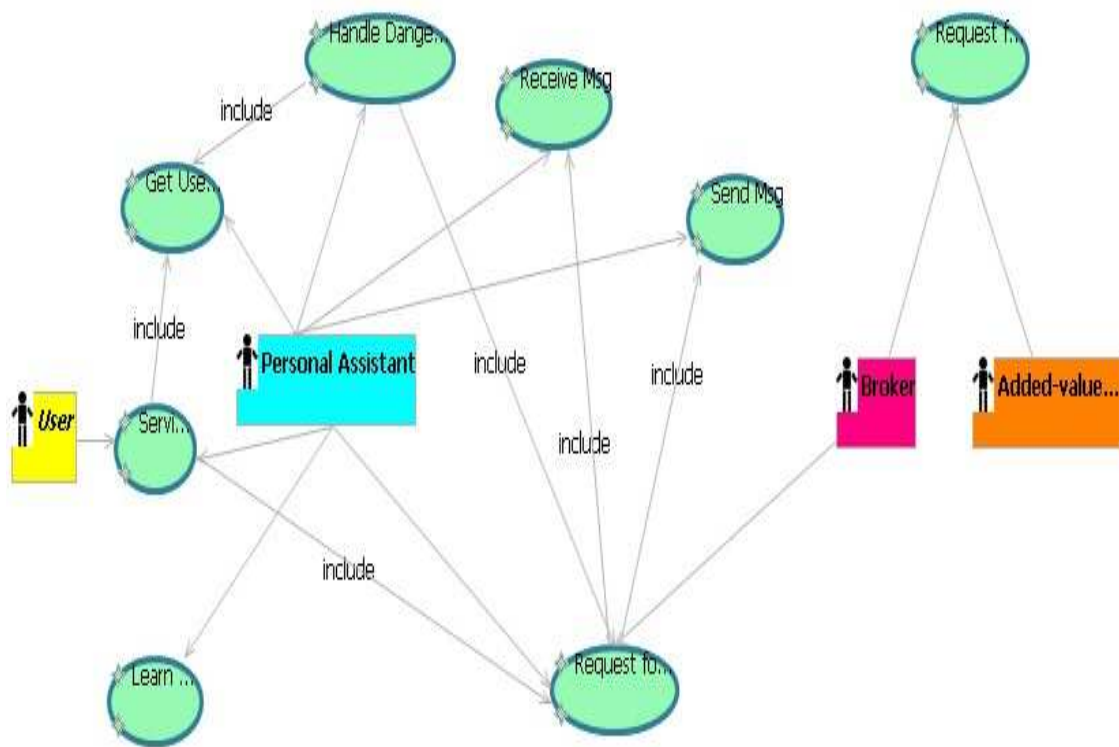
Είτε πατώντας κλικ στην επιλογή *transform* (στο ταμπλό GMF-dashboard) μεταξύ *Mapping Model* και *Diagram Editor Gen Model* και ακολουθώντας τον οδηγό, είτε με δεξί κλικ στο *SUC.gmfmap* αρχείο → *Create Generator Model* → *name* : *SUC.gmfgen* → ... → *Finish*, παίρνουμε ένα μοντέλο γεννήτριας για το διάγραμμά μας.

Στην συνέχεια δημιουργούμε τον κώδικα του συντάκτη διαγράμματος είτε ανοίγοντας το *gmfgen* μοντέλο και δεξί κλικ σ' αυτό → *Generate Diagram Code* ή το ίδιο στο *gmf* ταμπλό (dashboard). Δημιουργείται έτσι ένα νέο project στον χώρο εργασίας μας, το *SUCdesign.diagram*. Για να τρέξουμε τον συντάκτη διαγράμματος κάνουμε δεξί κλικ στο project αυτό και επιλέγουμε *Run AS-> Eclipse Application*. Για να κατασκευάσουμε ένα νέο διάγραμμα, δημιουργούμε ένα νέο project στο νέο στιγμιότυπο του *Eclipse*, επιλέγουμε *New* → *Other* και θα βρούμε το *SUC Diagram* ως επιλογή κάτω από τον φάκελο *Examples*. Ακολουθώντας τον οδηγό που εμφανίζεται κατασκευάζουμε το νέο μας διάγραμμα.



Μπορούμε να κατασκευάζουμε νέα διαγράμματα χωρίς να ανοίγουμε ένα νέο στιγμιότυπο του Eclipse , αλλά μέσα από το αρχικό στιγμιότυπο.Πρός τούτο, κάνουμε δεξί κλικ πάνω στο νέο plug-in (project) που δημιουργήσαμε (SUCdesign.diagram) → Export→Deployable plug-ins and fragments→ επιλογή plug-ins SUCdesign – SUCdesign.diagram-SUCdesign.edit & SUCdesign.editor→επιλογή φακέλου προορισμού→Finish

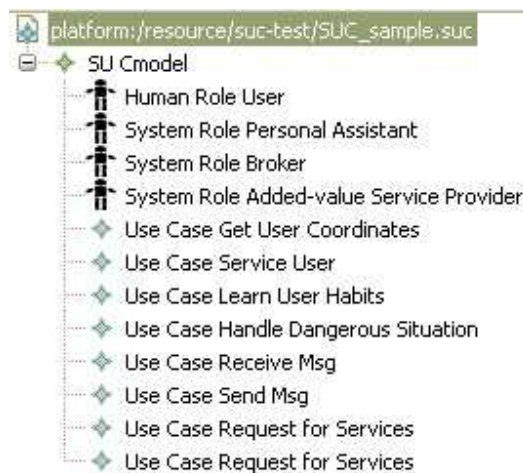
Τελικά αντιγράφουμε τα παραγόμενα . jar αρχεία στον φάκελο Eclipse plug-ins και ξανα-ανοίγουμε το Eclipse.Μπορούμε τώρα να κατασκευάσουμε μοντέλα SUC επιλέγοντας File→New→Other και ψάχνοντας για το διάγραμμα SAG.Με αυτό τον τρόπο κατασκευάσαμε τα αρχεία SUC_sample.sucd & SUC_sample.suc , για τις ανάγκες της εργασίας αυτής (βλ. εικ. 48 & 49).



Εικ. 48. Το νέο SUC μοντέλο (όνομα αρχείου SUC_sample.sucd)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 49. Το νέο Domain SUC μοντέλο (όνομα αρχείου SUC_sample.suc)

3.2.3 Γραφικό μοντέλο επεξεργασίας AIP διαγραμμάτων

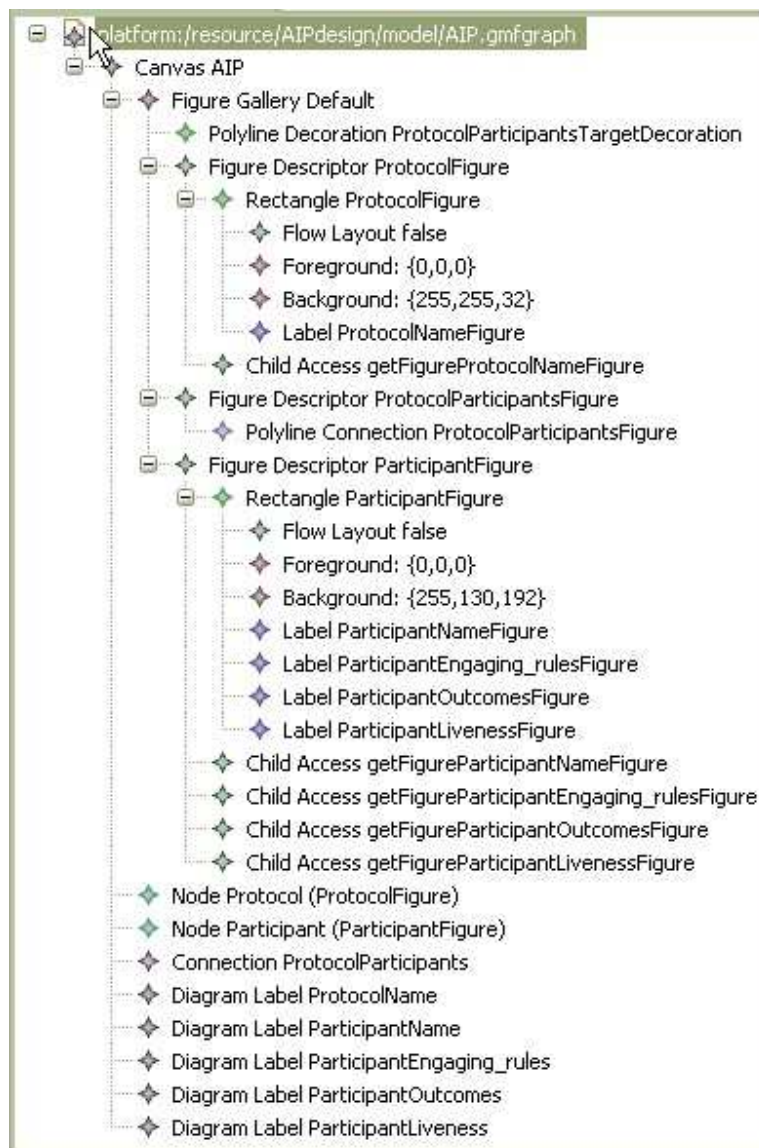
Ανάπτυξη του μοντέλου γραφικού καθορισμού (Graphical Definition Model)

Αρχίζουμε την ανάπτυξη του μοντέλου γραφικού καθορισμού είτε κλικαροντας στο *Derive* σε *Domain Model* και *Graphical Def Model* στο GMF ταμπλό (dashboard) ή απλά δεξί κλικ στον φάκελο *model* και επιλέγοντας *New* ➔ *Other* ➔ *Graphical Modeling Framework* ➔ *Simple Graphical Definition Model* ➔ *name:AIP.gmfgraph* ➔ *Next* ➔ *Load core file* ➔ *Diagram element : AIPmodel* ➔ επιλογή στοιχείων που θα παραχθούν στην αρχική έκδοση του γραφικού μοντέλου ➔ *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 50.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



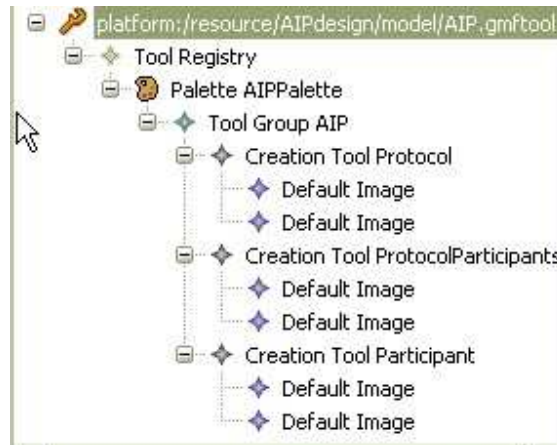
Εικ. 50. Το AIP Graphical Definition Model

Ανάπτυξη του μοντέλου καθορισμού σχεδίασης(Tooling Definition Model)

Ακολουθεί η δημιουργία των εργαλείων που θα χρησιμοποιήσουμε για να δημιουργήσουμε κλάσεις και σχέσεις στο διάγραμμα.Είτε αρχίζοντας από το GMF ταμπλό (dashboard) είτε με *New ➔Other ➔Graphical Modeling Framework ➔Simple Tooling Definition Model ➔name : AIP.gmftool ➔Next ➔Load Ecore file ➔Next ➔... ➔επιλογή στοιχείων (κλάσεων και σχέσεων) ➔Finish*



Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 51.



Εικ. 51. Το AIP Tooling Definition Model

Ανάπτυξη του μοντέλου καθορισμού χαρτογράφησης (Mapping Definition Model)

Ως τελικό βήμα στην δημιουργία του συγκεκριμένου συντάκτη , θα ενώσουμε το Ecore μεταμοντέλο , το γραφικό μοντέλο και το σχεδιαστικό μοντέλο μαζί σε ένα τρίτο GMF μοντέλο , το μοντέλο καθορισμού χαρτογράφησης. Αρχίζοντας όπως πάντα είτε από το GMF ταμπλό (dashboard) είτε με *New → Other → Graphical Modeling Framework → Guide Mapping Model Creation → name : AIP.gmfmap →* μέσω του οδηγού και κάνοντας τις κατάλληλες επιλογές θα καταλήξουμε να πάρουμε το παρακάτω αρχείο σε εκτεταμένη μορφή εικ. 52.



Εικ. 52. Το AIP Mapping Definition Model



Δημιουργώντας και εκτελώντας το διάγραμμα συντάκτη (Generating & Running the Diagram editor)

Είτε πατώντας κλικ στην επιλογή *transform* (στο ταμπλό *GMF-dashboard*) μεταξύ *Mapping Model* και *Diagram Editor Gen Model* και ακολουθώντας τον οδηγό, είτε με δεξί κλικ στο *AIP.gmfmap* αρχείο → *Create Generator Model* → *name* : *AIP.gmfgen* → ... → *Finish*, παίρνουμε ένα μοντέλο γεννήτριας για το διάγραμμά μας.

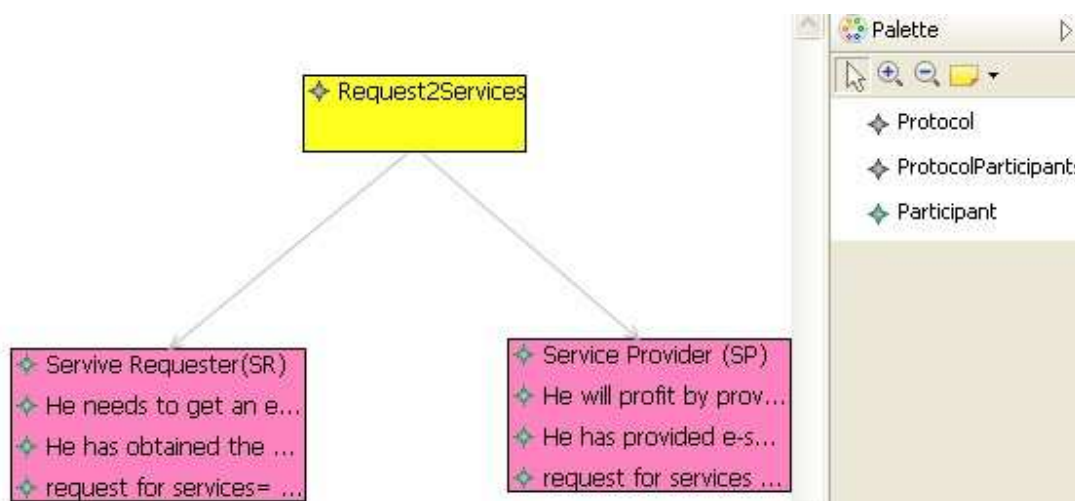
Στην συνέχεια δημιουργούμε τον κώδικα του συντάκτη διαγράμματος είτε ανοίγοντας το *gmfgen* μοντέλο και δεξί κλικ σ' αυτό → *Generate Diagram Code* ή το ίδιο στο *gmf* ταμπλό (*dashboard*). Δημιουργείται έτσι ένα νέο project στον χώρο εργασίας μας, το *AIPdesign.diagram*. Για να τρέξουμε τον συντάκτη διαγράμματος κάνουμε δεξί κλικ στο project αυτό και επιλέγουμε *Run AS-> Eclipse Application*. Για να κατασκευάσουμε ένα νέο διάγραμμα, δημιουργούμε ένα νέο project στο νέο στιγμιότυπο του *Eclipse*, επιλέγουμε *New* → *Other* και θα βρούμε το *AIP Diagram* ως επιλογή κάτω από τον φάκελο *Examples*. Ακολουθώντας τον οδηγό που εμφανίζεται κατασκευάζουμε το νέο μας διάγραμμα.

Μπορούμε να κατασκευάζουμε νέα διαγράμματα χωρίς να ανοίγουμε ένα νέο στιγμιότυπο του *Eclipse*, αλλά μέσα από το αρχικό στιγμιότυπο. Προς τούτο, κάνουμε δεξί κλικ πάνω στο νέο *plug-in* (project) που δημιουργήσαμε (*AIPdesign.diagram*) → *Export* → *Deployable plug-ins and fragments* → επιλογή *plug-ins AIPdesign - AIPdesign.diagram-AIPdesign.edit & AIPdesign.editor* → επιλογή φακέλου προορισμού → *Finish*

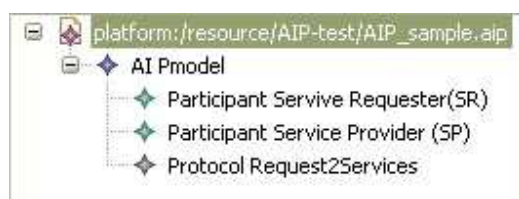
Τελικά αντιγράφουμε τα παραγόμενα .jar αρχεία στον φάκελο *Eclipse plug-ins* και ξανα-ανοίγουμε το *Eclipse*. Μπορούμε τώρα να κατασκευάσουμε μοντέλα *AIP* επιλέγοντας *File* → *New* → *Other* και ψάχνοντας για το διάγραμμα *AIP*. Με αυτό τον τρόπο κατασκευάσαμε τα αρχεία *AIP_sample.aipd* & *AIP_sample.aip*, για τις ανάγκες της εργασίας αυτής (βλ. εικ. 53 & 54).



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 53. Το νέο AIP μοντέλο (όνομα αρχείου AIP_sample.aipd)



Εικ. 54. Το νέο Domain AIP μοντέλο (όνομα αρχείου AIP_sample.aip)

3.2.4 Γραφικό μοντέλο επεξεργασίας SRM διαγραμμάτων

Ανάπτυξη του μοντέλου γραφικού καθορισμού (Graphical Definition Model)

Αρχίζουμε την ανάπτυξη του μοντέλου γραφικού καθορισμού είτε κλικαροντας στο *Derive* σε *Domain Model* και *Graphical Def Model* στο GMF ταμπλό (dashboard) ή απλά δεξί κλικ στον φάκελο *model* και επιλέγοντας *New* → *Other* → *Graphical Modeling Framework* → *Simple Graphical Definition Model* → *name:SRM.gmfgraph* → *Next* → *Load ecore file* → *Diagram element : SRMmodel* → επιλογή στοιχείων που θα παραχθούν στην αρχική έκδοση του γραφικού μοντέλου → Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 55.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



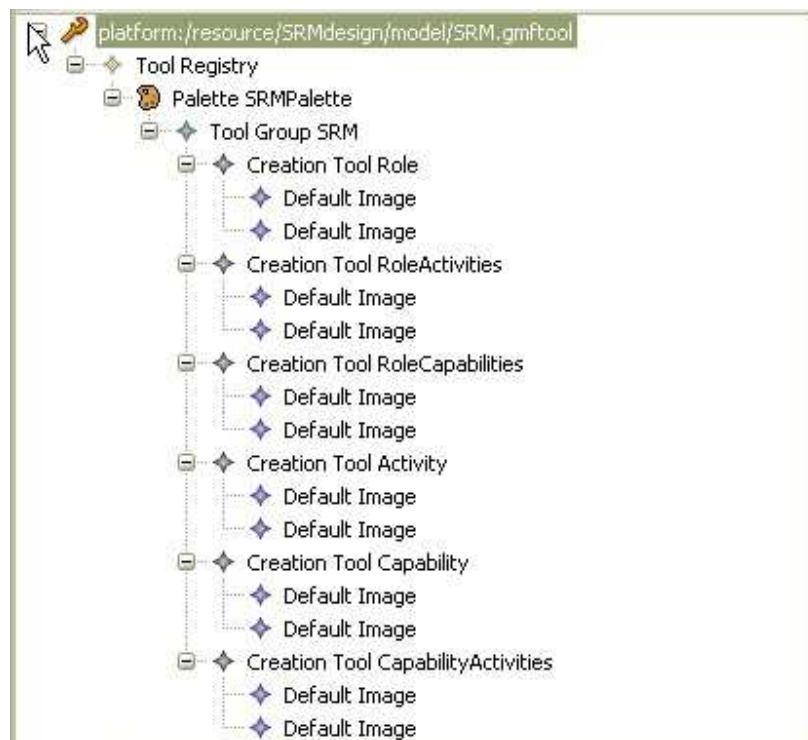
Εικ. 55. Το SRM Graphical Definition Model

Ανάπτυξη του μοντέλου καθορισμού σχεδίασης(Tooling Definition Model)



Ακολουθεί η δημιουργία των εργαλείων που θα χρησιμοποιήσουμε για να δημιουργήσουμε κλάσεις και σχέσεις στο διάγραμμα.Είτε αρχίζοντας από το GMF ταμπλό (dashboard) είτε με *New* → *Other* → *Graphical Modeling Framework* → *Simple Tooling Definition Model* → *name : SRM.gmftool* → *Next* → *Load Ecore file* → *Next* → ... → επιλογή στοιχείων (κλάσεων και σχέσεων) → *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 56.



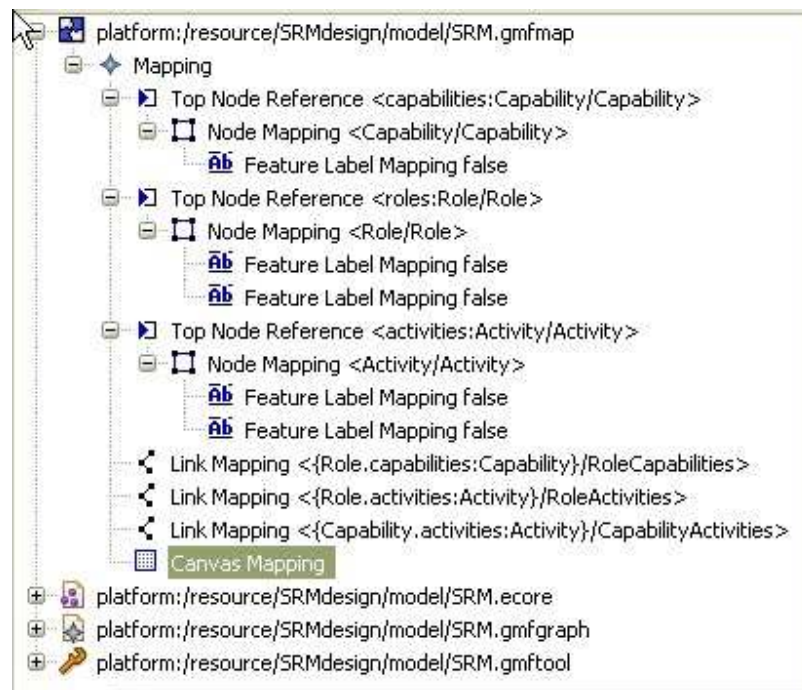
Εικ. 56. Το SRM Tooling Definition Model

Ανάπτυξη του μοντέλου καθορισμού χαρτογράφησης (Mapping Definition Model)

Ως τελικό βήμα στην δημιουργία του συγκεκριμένου συντάκτη , θα ενώσουμε το Ecore μεταμοντέλο , το γραφικό μοντέλο και το σχεδιαστικό μοντέλο μαζί σε ένα τρίτο GMF μοντέλο , το μοντέλο καθορισμού χαρτογράφησης.Αρχίζοντας όπως πάντα είτε από το GMF ταμπλό (dashboard) είτε με *New* → *Other* → *Graphical Modeling Framework* → *Guide Mapping Model Creation* → *name : SRM.gmfmap* →μέσω του οδηγού και κάνοντας τις κατάλληλες επιλογές θα καταλήξουμε να πάρουμε το παρακάτω αρχείο σε εκτεταμένη μορφή εικ. 57.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 57. Το SRM Mapping Definition Model

Δημιουργώντας και εκτελώντας το διάγραμμα συντάκτη (Generating & Running the Diagram editor)

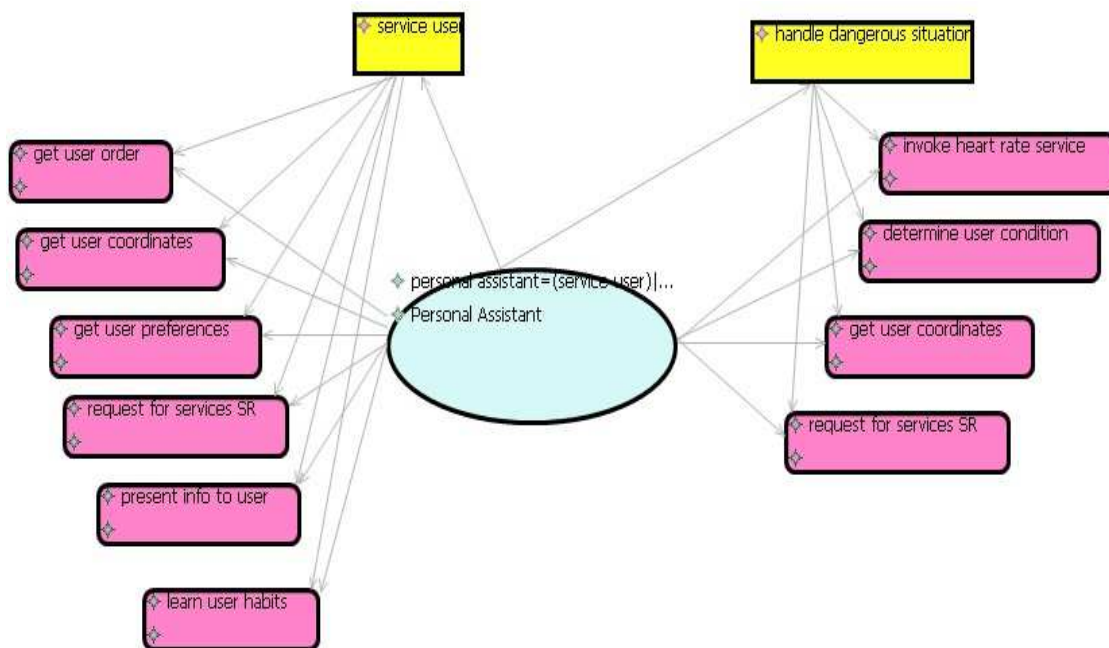
Είτε πατώντας κλικ στην επιλογή *transform* (στο ταμπλό GMF-dashboard) μεταξύ *Mapping Model* και *Diagram Editor Gen Model* και ακολουθώντας τον οδηγό, είτε με δεξί κλικ στο *SRM.gmfmap* αρχείο ➔ *Create Generator Model* ➔ *name : SRM.gmfgen* ➔ ... ➔ *Finish*, παίρνουμε ένα μοντέλο γεννήτριας για το διαγράμμα μας.

Στην συνέχεια δημιουργούμε τον κώδικα του συντάκτη διαγράμματος είτε ανοίγοντας το *gmfgen* μοντέλο και δεξί κλικ σ' αυτό ➔ *Generate Diagram Code* ή το ίδιο στο *gmf* ταμπλό (dashboard). Δημιουργείται έτσι ένα νέο project στον χώρο εργασίας μας, το *SRMdesign.diagram*. Για να τρέξουμε τον συντάκτη διαγράμματος κάνουμε δεξί κλικ στο project αυτό και επιλέγουμε *Run AS-> Eclipse Application*. Για να κατασκευάσουμε ένα νέο διάγραμμα, δημιουργούμε ένα νέο project στο νέο στιγμιότυπο του *Eclipse*, επιλέγουμε *New ➔ Other* και θα βρούμε το *SRM Diagram* ως επιλογή κάτω από τον φάκελο *Examples*. Ακολουθώντας τον οδηγό που εμφανίζεται κατασκευάζουμε το νέο μας διάγραμμα.



Μπορούμε να κατασκευάζουμε νέα διαγράμματα χωρίς να ανοίγουμε ένα νέο στιγμιότυπο του Eclipse , αλλά μέσα από το αρχικό στιγμιότυπο.Πρός τούτο, κάνουμε δεξί κλικ πάνω στο νέο plug-in (project) που δημιουργήσαμε (SRMdesign.diagram) → Export→Deployable plug-ins and fragments→ επιλογή plug-ins SRMdesign – SRMdesign.diagram-SRMdesign.edit & SRMdesign.editor→επιλογή φακέλου προορισμού→Finish

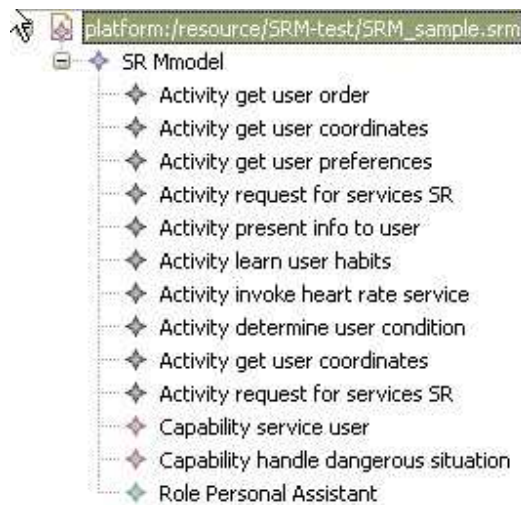
Τελικά αντιγράφουμε τα παραγόμενα . jar αρχεία στον φάκελο Eclipse plug-ins και ξανα-ανοίγουμε το Eclipse.Μπορούμε τώρα να κατασκευάσουμε μοντέλα SRM επιλέγοντας File→New→Other και ψάχνοντας για το διάγραμμα SRM.Με αυτό τον τρόπο κατασκευάσαμε τα αρχεία SRM_sample.srmd & SRM_sample.srm, για τις ανάγκες της εργασίας αυτής (βλ. εικ. 58 & 59).



Εικ. 58. Το νέο SRM μοντέλο (όνομα αρχείου SRM_sample.srmd)



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 59. Το νέο Domain SRM μοντέλο (όνομα αρχείου SRM_sample.srm)

3.2.5 Γραφικό μοντέλο επεξεργασίας IAC διαγραμμάτων

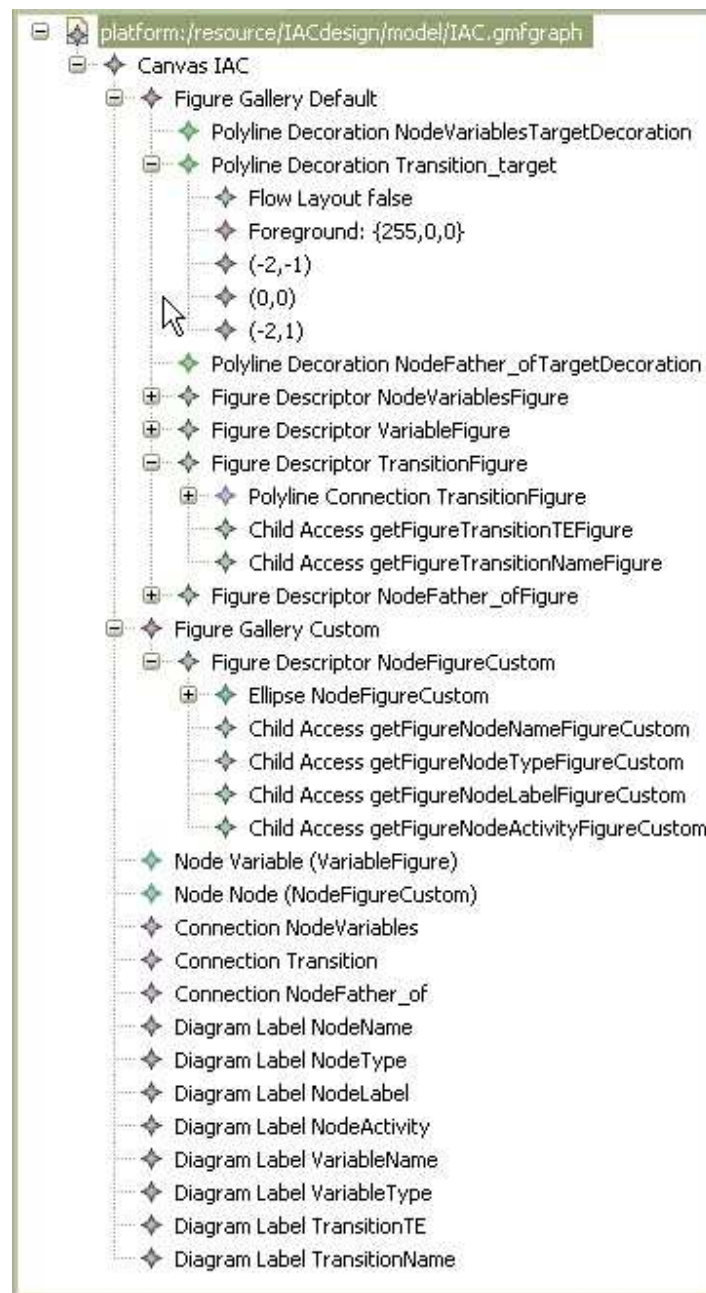
Ανάπτυξη του μοντέλου γραφικού καθορισμού (Graphical Definition Model)

Αρχίζουμε την ανάπτυξη του μοντέλου γραφικού καθορισμού είτε κλικαροντας στο *Derive* σε *Domain Model* και *Graphical Def Model* στο GMF ταμπλό (dashboard) ή απλά δεξί κλικ στον φάκελο *model* και επιλέγοντας *New* → *Other* → *Graphical Modeling Framework* → *Simple Graphical Definition Model* → *name:IAC.gmfgraph* → *Next* → *Load ecore file* → *Diagram element : IACmodel* → επιλογή στοιχείων που θα παραχθούν στην αρχική έκδοση του γραφικού μοντέλου → *Finish*

Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 60.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



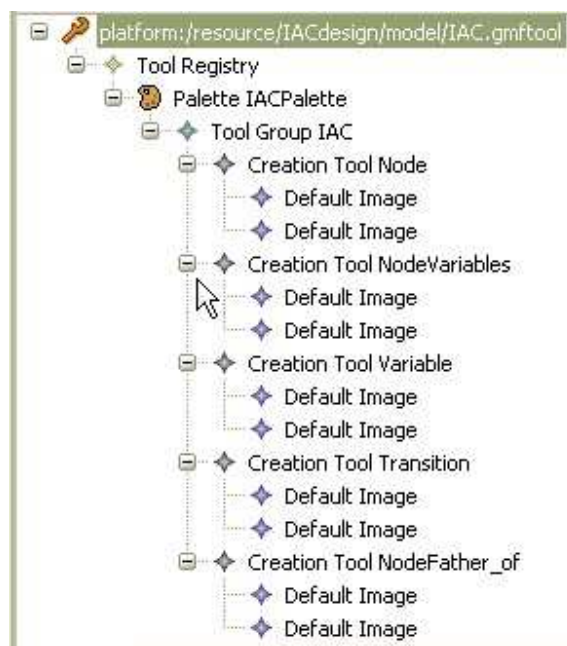
Εικ. 60. Το IAC Graphical Definition Model



Ανάπτυξη του μοντέλου καθορισμού σχεδίασης(Tooling Definition Model)

Ακολουθεί η δημιουργία των εργαλείων που θα χρησιμοποιήσουμε για να δημιουργήσουμε κλάσεις και σχέσεις στο διάγραμμα.Είτε αρχίζοντας από το GMF ταμπλό (dashboard) είτε με *New* ➔ *Other* ➔ *Graphical Modeling Framework* ➔ *Simple Tooling Definition Model* ➔ *name : IAC.gmftool* ➔ *Next* ➔ *Load Ecore file* ➔ *Next* ➔ ... ➔ επιλογή στοιχείων (κλάσεων και σχέσεων) ➔ *Finish*

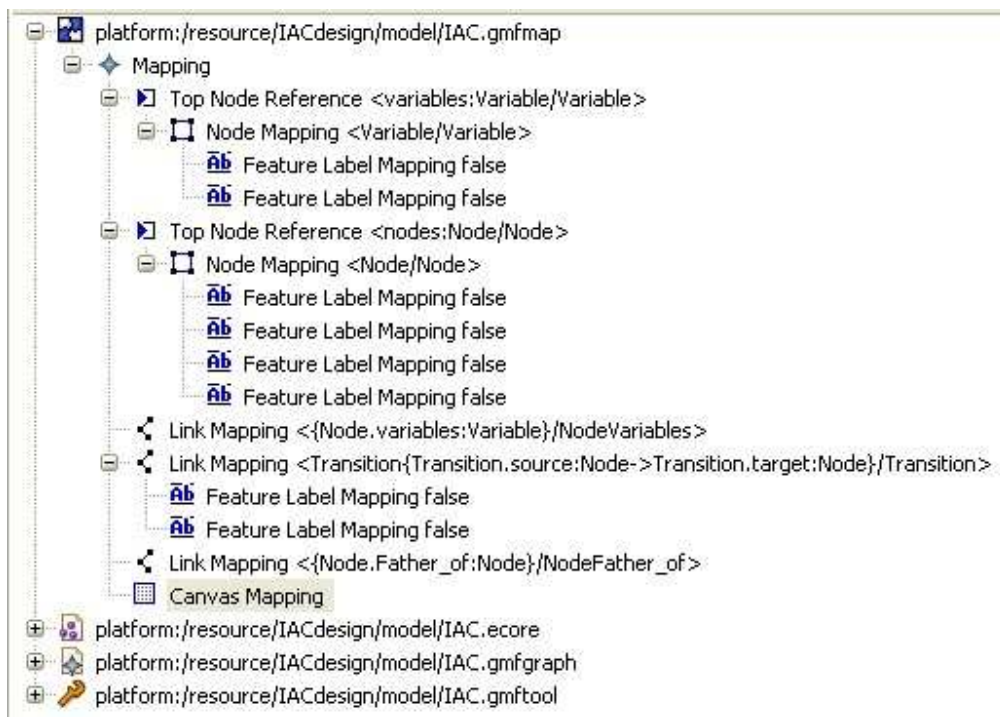
Το αποτέλεσμα θα είναι το ακόλουθο εκτεταμένο μοντέλο στην εικ. 61.



Εικ. 61. Το IAC Tooling Definition Model

Ανάπτυξη του μοντέλου καθορισμού χαρτογράφησης (Mapping Definition Model)

Ως τελικό βήμα στην δημιουργία του συγκεκριμένου συντάκτη , θα ενώσουμε το Ecore μεταμοντέλο , το γραφικό μοντέλο και το σχεδιαστικό μοντέλο μαζί σε ένα τρίτο GMF μοντέλο , το μοντέλο καθορισμού χαρτογράφησης.Αρχίζοντας όπως πάντα είτε από το GMF ταμπλό (dashboard) είτε με *New* ➔ *Other* ➔ *Graphical Modeling Framework* ➔ *Guide Mapping Model Creation* ➔ *name : IAC.gmfmap* ➔μέσω του οδηγού και κάνοντας τις κατάλληλες επιλογές θα καταλήξουμε να πάρουμε το παρακάτω αρχείο σε εκτεταμένη μορφή εικ. 62.



Εικ. 62. Το IAC Mapping Definition Model

Δημιουργώντας και εκτελώντας το διάγραμμα συντάκτη (Generating & Running the Diagram editor)

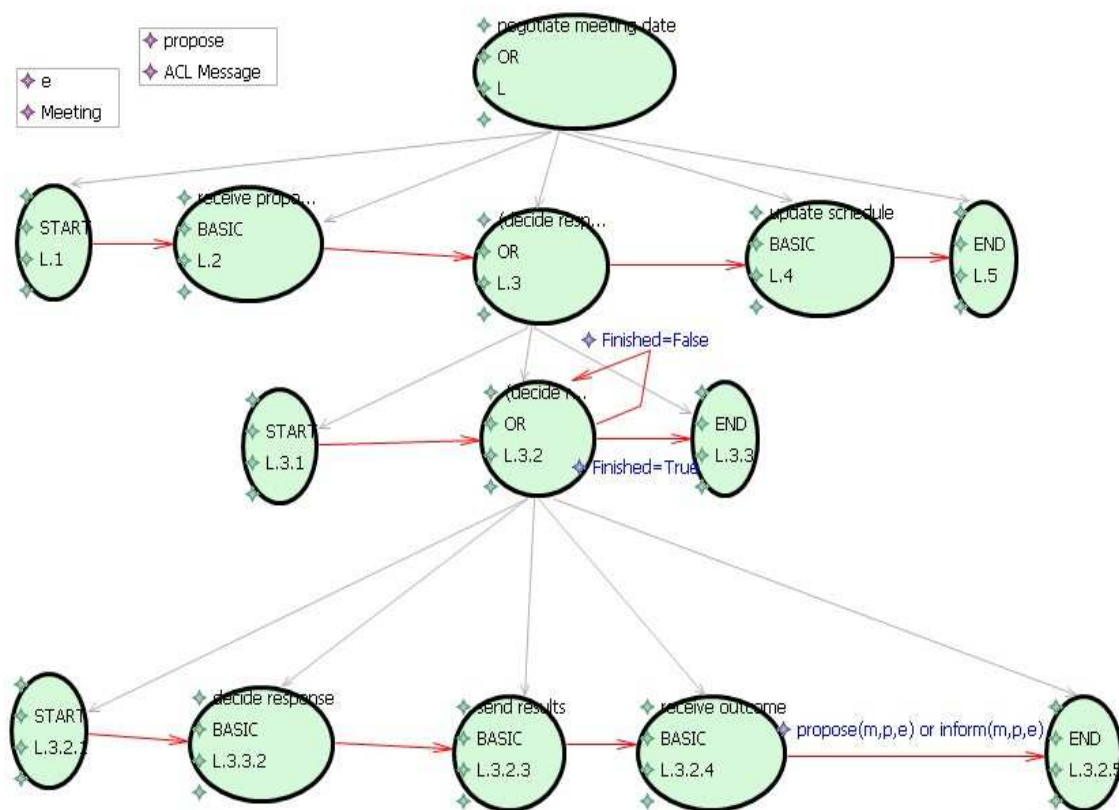
Είτε πατώντας κλικ στην επιλογή *transform* (στο ταμπλό GMF-dashboard) μεταξύ *Mapping Model* και *Diagram Editor Gen Model* και ακολουθώντας τον οδηγό, είτε με δεξί κλικ στο *IACgmfmap* αρχείο → *Create Generator Model* → *name* : *IAC.gmfgen* → ... → *Finish*, παίρνουμε ένα μοντέλο γεννήτριας για το διάγραμμά μας.

Στην συνέχεια δημιουργούμε τον κώδικα του συντάκτη διαγράμματος είτε ανοίγοντας το *gmfgen* μοντέλο και δεξί κλικ σ' αυτό → *Generate Diagram Code* ή το ίδιο στο *gmf* ταμπλό (dashboard). Δημιουργείται έτσι ένα νέο project στον χώρο εργασίας μας, το *IACdesign.diagram*. Για να τρέξουμε τον συντάκτη διαγράμματος κάνουμε δεξί κλικ στο project αυτό και επιλέγουμε *Run AS -> Eclipse Application*. Για να κατασκευάσουμε ένα νέο διάγραμμα, δημιουργούμε ένα νέο project στο νέο στιγμιότυπο του *Eclipse*, επιλέγουμε *New* → *Other* και θα βρούμε το *IAC Diagram* ως επιλογή κάτω από τον

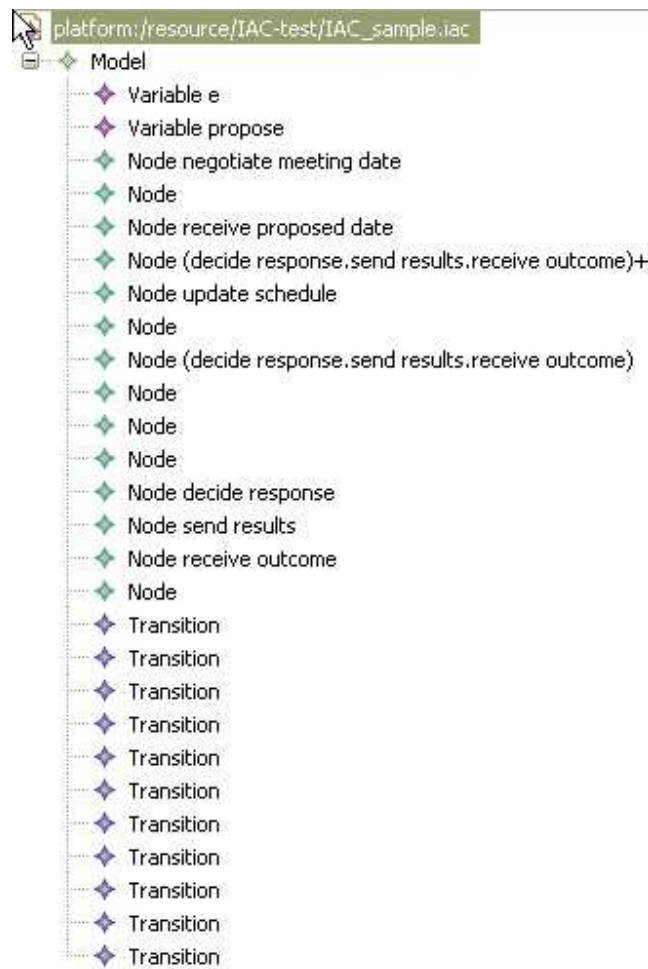


φάκελο Examples. Ακολουθώντας τον οδηγό που εμφανίζεται κατασκευάζουμε το νέο μας διάγραμμα.

Μπορούμε να κατασκευάζουμε νέα διαγράμματα χωρίς να ανοίγουμε ένα νέο στιγμιότυπο του Eclipse, αλλά μέσα από το αρχικό στιγμιότυπο. Προς τούτο, κάνουμε δεξί κλικ πάνω στο νέο plug-in (project) που δημιουργήσαμε (IACdesign.diagram) → Export → Deployable plug-ins and fragments → επιλογή plug-ins IACdesign – IACdesign.diagram – IACdesign.edit & IACdesign.editor → επιλογή φακέλου προορισμού → Finish. Τελικά αντιγράφουμε τα παραγόμενα .jar αρχεία στον φάκελο Eclipse plug-ins και ξανα-ανοίγουμε το Eclipse. Μπορούμε τώρα να κατασκευάσουμε μοντέλα IAC επιλέγοντας File → New → Other και ψάχνοντας για το διάγραμμα IAC. Με αυτό τον τρόπο κατασκευάσαμε τα αρχεία IAC_sample.iacd & IAC_sample.iac, για τις ανάγκες της εργασίας αυτής (βλ. εικ. 63 & 64).



Εικ. 63. Το νέο IAC μοντέλο (όνομα αρχείου IAC_sample.iacd). Δενδρική αναπαράσταση του διαγράμματος καταστάσεων



Εικ. 64. Το νέο Domain IAC μοντέλο (όνομα αρχείου IAC_sample.iac)

Το δένδρο του διαγράμματος καταστάσεων ορίζει μια ιεραρχία κόμβων οι οποίοι αντιστοιχούν σε καταστάσεις και σε υπο-καταστάσεις. Το διάγραμμα περιέχει επίσης μεταβάσεις, τόσο υπό συνθήκη όσο και χωρίς συνθήκη. Οι υπό συνθήκη μεταβάσεις περιέχουν *λογικές εκφράσεις* που σχετίζονται με τον αντίστοιχο κόμβο. Η δομή των κόμβων καταστάσεων είναι ίδια με την δομή των *activities*.



4. Μετασχηματισμοί στην ASEME

Οι μετασχηματισμοί που δημιουργήθηκαν και εκτελέστηκαν για τις ανάγκες της εργασίας αυτής, ήταν μετασχηματισμοί μοντέλου σε μοντέλο (Model to Model -M2M Transformation) και χρησιμοποιήθηκε η γλώσσα μετασχηματισμού μοντέλων ATLAS. Τα γενικά και ειδικά χαρακτηριστικά των μετασχηματισμών M2M, καθώς και της γλώσσας μετασχηματισμού αναφέρθηκαν στην §2.2.2

4.1 Μετασχηματισμός SAG2SUC

Εδώ έγινε μια τροποποίηση του αρχείου μετασχηματισμού SAG2SUC (δηλ. το SAG2SUC.atl) που αναφέρθηκε στην §2.2.2, από το γεγονός ότι τροποποιήθηκαν τα αρχεία των μεταμοντέλων SAG και SUC (σε ecore φορμά). Προς τούτο προστέθηκε ένας επιπλέον κανόνας Model2Model, με σκοπό να εξυπηρετήσει την αλλαγή αυτή. Το νέο αρχείο μετασχηματισμού SAG2SUC.atl παρουσιάζεται στην εικ. 65.

```
-- @path SAG=/ASEME-ATL_ARGIRIOU/metamodels/SAG.ecore
-- @path SUC=/ASEME-ATL_ARGIRIOU/metamodels/SUC.ecore

module SAG2SUC; -- Module Template
create OUT : SUC from IN : SAG;

rule SAGmodel2SUCmodel {
  from
  i : SAG!SAGmodel
  to
  o_1:SUC!SUCmodel {
    roles <- i.actors,
    usecases <- i.goals
  }
}

rule Goal2UseCase {
  from
  i : SAG!Goal
  to
  o_1 : SUC!UseCase {
    name <- i.name,
    specified_by <- i.requirements,
    participant <- i.depender,
    participant <- i.dependee
  }
}

rule Actor2Role {
  from
  i : SAG!Actor
  to
  o_n : SUC!Role {
    name <- i.name
  }
}
```

Εικ. 65. Το αρχείο του μετασχηματισμού SAG2SUC(SAG2SUC.atl)



Για να εκτελέσουμε τον μετασχηματισμό , ο τύπος του ATL μετασχηματισμού στο Eclipse Run Configurations πρέπει να στιγμιτυποποιηθεί.Εμφανίζει ένα παράθυρο διαλόγου όπου επιλέγεται ο τύπος των μεταμοντέλων (στην περίπτωση EMF) παράλληλα με τα αρχεία μεταμοντέλων και μοντέλων (βλ. εικ. 66). Αν χρησιμοποιηθούν τα ονόματα π.χ. για τα μοντέλα SAG_sample.xmi και SUC_sample.xmi, αυτόματα μπορεί να προκληθεί μετασχηματισμός εκτελώντας τον ASEME_SAG2SUC μετασχηματισμό.



Εικ. 66. Run Configuration του ATL μετασχηματισμού SAG2SUC του Eclipse



4.2 Μετασχηματισμός SUC+AIP2SRM

Εδώ έγινε αφ' ενός μια τροποποίηση του αρχείου μετασχηματισμού SUC2SRM (το SUC2SRM.atl) που αναφέρθηκε στην §2.2.2 , από το γεγονός ότι τροποποιήθηκαν τα αρχεία των μεταμοντέλων SUC και SRM (σε ecore φορμά) δηλ. προστέθηκαν δύο επιπλέον κανόνες Model2Model , με σκοπό να εξυπηρετήσει την αλλαγή αυτή , αφ' ετέρου και εμπλουτισμός του ίδιου μετασχηματισμού ώστε να δημιουργείται ένα πρώτο liveness σε κάθε ρόλο με τη μεταφορά των κατάλληλων πεδίων process από το AIP στο SRM. Με αυτό τον τρόπο δημιουργείται ένα νέο αρχείο μετασχηματισμού , το SUC_AIP2SRM.atl που παρουσιάζεται στην εικ. 67.

```
-- @path SRM=/ASEME-ATL_ARGIRIOU/metamodels/SRM.ecore
-- @path SUC=/ASEME-ATL_ARGIRIOU/metamodels/SUC.ecore
-- @path AIP=/ASEME-ATL_ARGIRIOU/metamodels/AIP.ecore
module SUC_AIP2SRM; -- Module Template
create OUT : SRM from IN : SUC , IN : AIP;

rule SUCmodel2SRMmodel {
    from
        i:SUC!SUCmodel
    to
        o_1 : SRM!SRMmodel {
            roles <i.roles ,
            capabilities <i.usecases,
            activities <i.usecases
        }
}

rule AIPmodel2SRMmodel {
    from
        i:AIP!AIPmodel
    to
        o_1 : SRM!SRMmodel {
            roles <i.participants
        }
}

rule Role2Role {
    from
        i : SUC!Role
    to
        o_1 : SRM!SystemRole {
            name <- i.name,
            activities <- Sequence {} -> union(i.participates_in->select(e | not(e.participant->size()>1))),
            capabilities <- Sequence {} -> union(i.participates_in->select(e | e.participant->size()>1))
        }
}

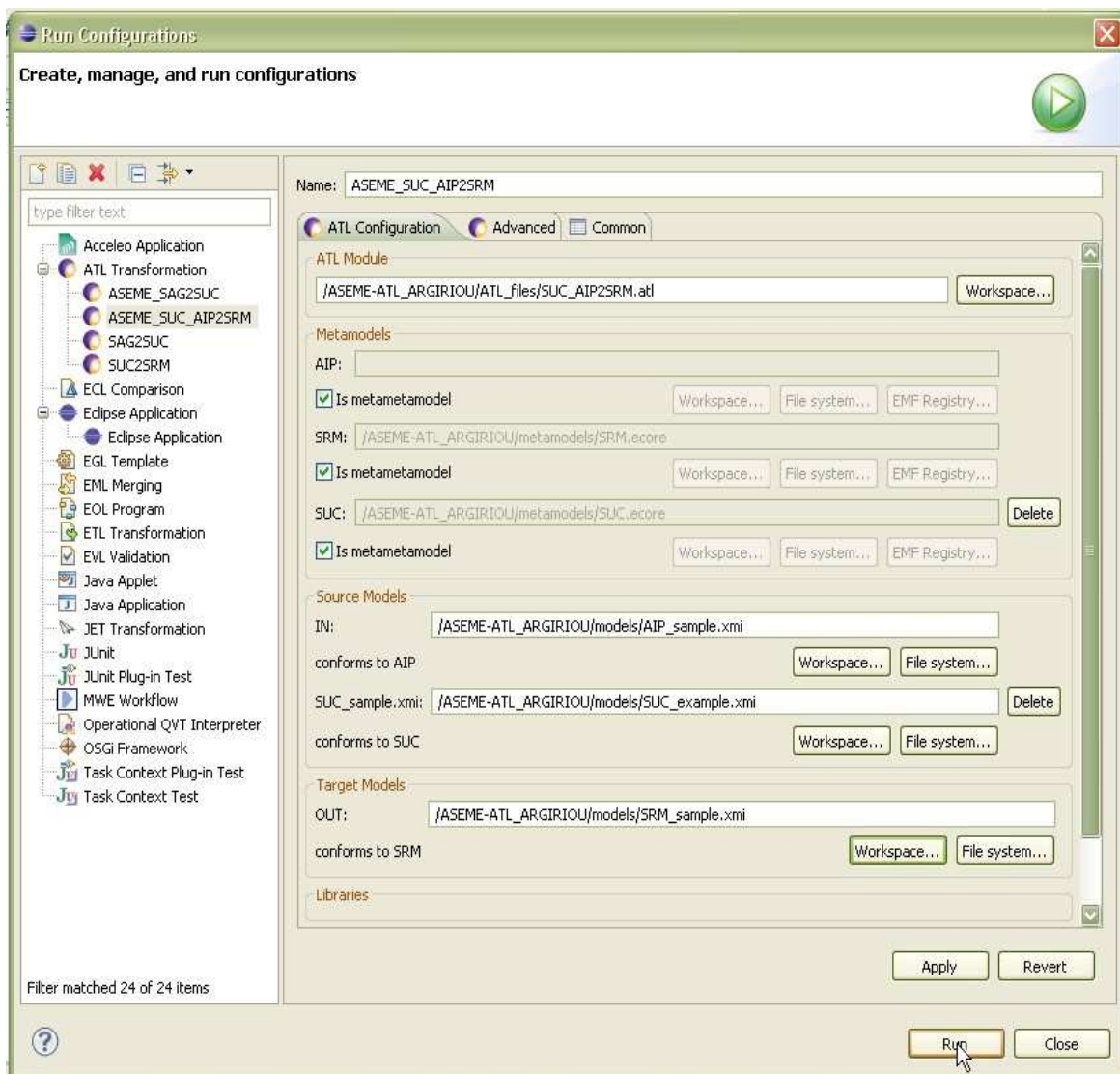
rule Participant2Role {
    from
        i : AIP!Participant
    to
        o_1 : SRM!SystemRole {
            liveness<- i.liveness
        }
}
```

Εικ. 67. Το αρχείο του μετασχηματισμού SUC+AIP2SRM(SUC_AIP2SRM.atl)

Για να εκτελέσουμε τον μετασχηματισμό , ο τύπος του ATL μετασχηματισμού στο Eclipse Run Configurations πρέπει να στιγμιτυποποιηθεί. Εμφανίζει ένα παράθυρο



διαλόγου όπου επιλέγεται ο τύπος των μεταμοντέλων (στην περίπτωση EMF) παράλληλα με τα αρχεία μεταμοντέλων και μοντέλων (βλ. εικ. 68). Αν χρησιμοποιηθούν τα ονόματα π.χ. για τα μοντέλα SUC_sample.xmi και AIP_sample.xmi, αυτόματα μπορεί να προκληθεί μετασχηματισμός εκτελώντας τον ASEME_SUC_AIP2SRM μετασχηματισμό, παίρνοντας ως έξοδο το αρχείο SRM_sample.xmi



Εικ. 68. Run Configuration του ATL μετασχηματισμού (SUC+AIP2SRM) του Eclipse



5. Συμπεράσματα – Αξιολογήσεις – Μελλοντικές κατευθύνσεις

Στα πλαίσια της παρούσας διπλωματικής εργασίας , έγινε μέσω της προσέγγισης γραφικής αναπαράστασης μοντέλων , η υλοποίηση των μετασχηματισμών μοντέλων (model to model transformation ή M2M transformation) , που ορίστηκαν στην μεθοδολογία ASEME , μια μοντέρνα μεθοδολογία η οποία βασίζεται σε μια Οδηγούμενη από Μοντέλα Αρχιτεκτονική (Model Driven Architecture-MDA) της γλώσσας μοντελοποίησης πρακτόρων AMOLA(Agent Modeling Language).Ως πράκτορας ορίζεται μια οντότητα με τις δικές της ικανότητες , συμπεριλαμβανομένης της ενδο- και δια-πρακτορικής επικοινωνίας [35].

Η υλοποίηση αυτή έγινε με χρήση εργαλείων γραφικής αναπαράστασης μοντέλων και μεταμοντέλων , τα οποία κατασκευάστηκαν μέσα από το ανοικτό ολοκληρωμένο περιβάλλον ανάπτυξης της πλατφόρμας του Eclipse χρησιμοποιώντας τεχνολογίες που είναι ενσωματωμένες σ' αυτό.Ως τεχνολογία μετασχηματισμού χρησιμοποιήθηκε η γλώσσα μετασχηματισμού ATLAS (Atlas Transformation Language –ATL) και ως τεχνολογίες κατασκευής και γραφικής αναπαράστασης μεταμοντέλων και μοντέλων το Eclipse Modeling Framework (EMF) και το Graphical Modeling Framework (GMF) , αντίστοιχα.

Ειδικότερα έγινε η θεωρητική μελέτη του περιβάλλοντος υλοποίησης των μετασχηματισμών μοντέλων για τις φάσεις ανάλυσης προδιαγραφών, ανάλυσης και σχεδίασης συστήματος της πρακτοροστραφούς μεθοδολογίας ανάπτυξης λογισμικού ASEME , καθώς επίσης των αναγκαίων τεχνολογιών τόσο για αυτούς καθ' αυτούς τους μετασχηματισμούς μοντέλων (γλώσσες μετασχηματισμών) όσο και για την γραφική αναπαράσταση αυτών (τεχνολογίες GMF, EuGENia , TAOM4E) και των μεταμοντέλων τους (EMF , MOF).

Από την πλευρά της υλοποίησης κατασκευάστηκαν οι γραφικοί συντάκτες για την μοντελοποίηση των μοντέλων αλλά και των μεταμοντέλων τους που ήταν αναγκαία, μέχρι την φάση της σχεδίασης των πρακτόρων της επιλεγείσας μεθοδολογίας ASEME



δηλ. των SAG (System Actor Goal) , SUC (System UseCase) , AIP (Agent Interaction Protocol), SRM (System Role Model) και IAC (Intra-Agent Control).

Ως προς τους μετασχηματισμούς , έγινε υλοποίηση των μετασχηματισμών για τις φάσεις ανάλυσης προδιαγραφών και ανάλυσης , SAG2SUC(μετασχηματίζοντας δράστες σε ρόλους και στόχους σε περιπτώσεις χρήσης) καθώς και SUC+AIP2SRM (μετασχηματίζοντας γενικές περιπτώσεις χρήσης σε ικανότητες και συγκεκριμένες δραστηριότητες καθώς και εμπλουτίζοντας τον ίδιο τον μετασχηματισμό ώστε να δημιουργείται ένα πρώτο liveness σε κάθε ρόλο με τη μεταφορά των κατάλληλων πεδίων process από το AIP στο SRM μοντέλο), μετά από τροποποίηση των μεταμοντέλων και των μοντέλων τους για τις ανάγκες της εργασίας αυτής με βάση το κοινό παράδειγμα εργασίας 'σύστημα διαχείρισης συναντήσεων' , που αναφέρθηκε στο θεωρητικό μέρος.

Κατά την εξέλιξη της εργασίας διαφάνηκε η σπουδαιότητα της μεθοδολογίας ASEME στην υλοποίηση προσθέτων του περιβάλλοντος Eclipse , τα οποία μπορούν να καθοδηγήσουν τον σχεδιαστή ενός συστήματος μέσω των βημάτων ανάπτυξης και να του παράξουν ωραία γραφικά εργαλεία , όπως οι συντάκτες *Sample Reflective Ecore Model* ή *Exceed* και τα graphical , tooling & mapping definition tools του GMF plug-in του Eclipse.

Προφανώς η ίδια μεθοδολογία θα χρησιμοποιηθεί , σε μελλοντική εργασία επεκτείνοντας την παρούσα , για την αυτοματοποίηση των μετασχηματισμών για τα μοντέλα των επόμενων φάσεων ανάπτυξης του συστήματος – μετασχηματισμοί M2T (model to text) και T2T (text to text) , δηλ. παραγωγή κώδικα (χρήση JADE Framework).Σαν επέκταση-συμπλήρωση της παρούσας εργασίας θα μπορούσε να δημιουργηθεί ένας οδηγός (με γραφικό περιβάλλον) που θα καθοδηγεί τον χρήστη στην ανάπτυξη μιας πλήρους εφαρμογής με την χρήση της ASEME.

Μια άλλη ενδιαφέρουσα δυνατότητα της ASEME είναι ότι θα μπορούσε να ορίσει τους M2M μετασχηματισμούς για τα μοντέλα των μεθοδολογιών Gaia , Tropos και UML έτσι ώστε τα υπάρχοντα μοντέλα συστήματος να στραφούν προς την AMOLA για περισσότερη επεξεργασία , χρησιμοποιώντας την ASEME διαδικασία.Με αυτό τον τρόπο θα μπορούσε να αυτοματοποιηθεί η διαδικασία ολοκλήρωσης ενός τμήματος μεθόδου των Gaia , Tropos ή UML στην ASEME μεθοδολογία.



Τα μοντέλα της AMOLA μπορούν να οδηγήσουν σε ανάπτυξη πράκτορα χωρίς περιορισμούς στην παρουσίαση του νοητικού του μοντέλου (σε αντίθεση με άλλες υπάρχουσες μεθοδολογίες π.χ. Ignenias και Prometheus).

Η ASEME ορίζει τρία καθαρά επίπεδα αφαίρεσης σ' όλες τις φάσεις ανάπτυξης , μια μοναδική πρωτοτυπία σε αντίθεση με όλες τις άλλες μεθοδολογίες.Επίσης ορίζει ένα διαφορετικό μετα-μοντέλο για κάθε φάση ανάπτυξης ενώ όλες οι άλλες μεθοδολογίες έχουν ένα μοναδικό μετα-μοντέλο που καλύπτει όλες τις φάσεις , σύνθετο και δύσκολο να κατανοηθεί τι πληροφορία προσθέτει σε κάθε φάση.Είναι η πρώτη πρακτοροστραφής μεθοδολογία δημιουργίας λογισμικού που περιλαμβάνει τρεις τύπους μετασχηματισμών (M2M , M2T και T2M) και χρησιμοποιεί επίσης μη λειτουργικές απαιτήσεις για να επηρεάσουν τον τύπο της αρχιτεκτονικής λύσης και τις τεχνολογίες που θα εφαρμοσθούν [10].Η μόνη μεθοδολογία που επιτρέπει μοντελοποίηση μη λειτουργικών απαιτήσεων είναι η Tropos, ορίζοντας γι' αυτό τους μαλακούς στόχους (soft goals) , αλλά δεν δείχνει τον τρόπο που αυτοί επηρεάζουν την διαδικασία ανάπτυξης.

Η ανάγκη διαδραστικότητας ανάμεσα στον άνθρωπο και τον υπολογιστή μπορεί αυτόματα να παραχθεί από τον τύπο των ρόλων του SUC μοντέλου.Κατόπιν οι διαφορετικές μεταβάσεις που βγαίνουν από μια περίπτωση χρήσης προς τον χρήστη μπορούν να αυτοματοποιηθούν σε διαφορετικές επιλογές για τον χρήστη , σε μια αυτόματα παραγόμενη μορφή.

Πέρα απο την ήδη αναφερθείσα δυνατότητα επέκτασης των μετασχηματισμών μοντέλων και για τις άλλες φάσεις ανάπτυξης ενός συστήματος λογισμικού , παραμένουν υπό αξιοποίηση οι ερευνητικές κατευθύνσεις χρήσης της εν λόγω μεθοδολογίας για αυτοματοποίηση ανάπτυξης λογισμικού και αυτοματοποίηση της διαδικασίας παραγωγής μοντέλων.

5.1 Μελλοντικές κατευθύνσεις της έρευνας

Αυτοματοποίηση ανάπτυξης λογισμικού

Μια από τις πιο ενδιαφέρουσες προκλήσεις είναι να καθορίσουμε την γνώση για εμπλουτισμό των μοντέλων σε κάθε φάση ανάπτυξης.Στο μοντέλο SUC, η γνώση θα μπορούσε να αναφέρεται στον τρόπο με τον οποίο θα αποσυνθέσουμε μια γενική



διεργασία σε άλλες πιο συγκεκριμένες.Επιπλέον οι συγκεκριμένες αυτές διεργασίες πρέπει να συνδεθούν με μια τεχνολογία για την επίτευξή τους.Στο μοντέλο SRM, η γνώση πρέπει να αναφέρεται στον τρόπο που θα συναρμολογήσουμε τις προηγούμενες διεργασίες ώστε να αποτελέσουν μια ικανότητα.Για το μοντέλο IAC , η γνώση πρέπει να αναφέρεται στον ορισμό των μεταβλητών και την τεκμηρίωση της λειτουργικότητας σε μια μορφή όσο πιο κοντά στον αναπτυσσόμενο κώδικα.Καθώς το IAC είναι ανεξάρτητο από πλατφόρμες , είναι δύσκολο να επισημάνουμε μια συγκεκριμένη γλώσσα για να περιγράψουμε την λειτουργικότητα , εν τούτοις θα μπορούσε να χρησιμοποιηθεί ένας ψευδοκώδικας ή υλοποίηση οποιαδήποτε πλατφόρμας.

Αυτοματοποίηση διαδικασίας παραγωγής μοντέλων

Μια σπουδαία κατεύθυνση έρευνας είναι προς την αυτοματοποίηση του μετασχηματισμού του μοντέλου IAC σε μοντέλο διαδικασίας.Το πρόβλημα με αυτό το ζήτημα είναι ότι αντίθετα με την κοινότητα ανάπτυξης λογισμικού , όπου μεταμοντέλα για διάφορους τύπους μοντέλων είναι γενικά τυποποιημένα , τα μοντέλα διαδικασίας δεν είναι διαδεδομένα και έτσι δεν υπάρχουν στην αγορά διαθέσιμα εργαλεία που να υποστηρίζουν ένα κοινό φορμά.Οπωσδήποτε μια περισσότερο εστιασμένη στην κατεύθυνση αυτή ερευνητική προσπάθεια , θα μπορούσε να φέρει κάτι σαν μεταμοντέλο διαδικασίας για το περιβάλλον του Eclipse , παράλληλα με εργαλεία προσομοίωσης όμοια με του εμπορίου [35].

Δημιουργία web-sites από μοντέλα

Ένα ζητούμενο είναι να μπορέσουμε να αποδείξουμε , αν μια γλώσσα ειδικού σκοπού (Domain Specific Language) είναι μια περισσότερο αποτελεσματική λύση στην δημιουργία **web-sites** από τα υπάρχοντα αναφαινόμενα εργαλεία δηλ. ότι η ασαφής σχέση μεταξύ μοντέλου και της αντίστοιχης υλοποίησης μπορεί να γίνει όλο και λιγότερο ασαφής χρησιμοποιώντας μια γλώσσα μόνο για αυτόν τον σκοπό – η χρήση των PHP και MySQL επιβάλλει περιορισμούς στην αρχιτεκτονική σχεδίαση.Ο ορισμός του μεταμοντέλου ενός web site θα μπορούσε να βελτιώσει περισσότερο τον ορισμό των συστατικών ενός web site.Η δημιουργία τέτοιων sites θα μπορούσε να γίνει μέσω model2website μετασχηματισμών , που θα είχαν ως είσοδο EMF και GMF μοντέλα και ως έξοδο παραγόμενο κώδικα (για M2T μετασχηματισμούς).



Μέτρηση της ποιότητας των μετασχηματισμών μοντέλων σε MDD (Model Driven Development) περιβάλλον

Στο περιβάλλον της Οδηγούμενης από Μοντέλα ανάπτυξης λογισμικού (MDD) , είναι σημαντικά ο μετασχηματισμός μοντέλων ως μια τεχνολογία-κλειδί και οι μετασχηματισμοί που μπορούν να βελτιώσουν την ποιότητα των μοντέλων.Ένα από τα προβλήματα της παραπάνω μεθοδολογίας είναι πώς να αναγνωρίζουμε τους μετασχηματισμούς που μπορούν να βελτιώσουν την ποιότητα των μοντέλων.Αν η τιμή μιας μετρικής μπορεί να εκφράσει την ποιότητα ενός μοντέλου , αυξάνοντας τις τιμές των μετρικής πριν και μετά τον μετασχηματισμό μοντέλου , μπορούμε να δούμε την βελτίωση της ποιότητας του μοντέλου.Αυτό σημαίνει ότι ο τυπικός ορισμός ενός μετασχηματισμού θα μπορούσε να περιλαμβάνει τον ορισμό των μετρικών της ποιότητας των μοντέλων , έτσι ώστε οι μετρικές να μπορούν να υπολογισθούν κατά την διάρκεια του μετασχηματισμού.Ως χαρακτηριστικά ποιότητας θα μπορούσαν να χρησιμοποιηθούν η κατανοησιμότητα (understandability) , η τροποποιησιμότητα (modifiability) , η επαναχρησιμοποιησιμότητα (reusability) , η δομησιμότητα (modularity) , η πληρότητα (completeness) , η συνέπεια (consistency) , η συνοπτικότητα (conciseness) κ.α. και ως μετρικές το μέγεθος (size) , ο αριθμός των συναρτήσεων (functions) , ο αριθμός των modules , για τους μετασχηματισμούς με την γλώσσα ATLAS ο αριθμός των χρησιμοποιούμενων κανόνων (transformation rules) κ.λ.π.

Επίσης χρήση της QVT ως γλώσσας μετασχηματισμού και σύγκριση της με την ATLAS, ως προς την αποτελεσματικότητά τους – με πιθανή συνδιασμένη χρήση μετρικών - θα μπορούσε να αποτελέσει θέμα μελλοντικής εργασίας.



Αναφορές

1. Bellifemine, F., A. Poggi, and G. Rimassa. Developing Multi-Agent Systems with JADE. In C. Castelfranchi and Y. Lesperance, editors, Intelligent Agents VII. Agent Theories, Architectures, and Languages-7th. International Workshop, ATAL-2000, Boston, MA, USA, July 7–9, 2000, Proceedings, Lecture Notes in Artificial Intelligence. Volume 1986/2001, Springer-Verlag, Berlin, 2001.
2. Bernon, C., Cossentino, M., Pavon, J.: (2005) Agent Oriented Software Engineering. The Knowledge Engineering Review, Cambridge University Press, 20: pp.99-116
3. Beydeda, S., Book, M., Gruhn, V.: Model-Driven Software Development. Springer (2005)
4. Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J. and Perini, A. 2004. TROPOS: An Agent-Oriented Software Development Methodology. J. Auton. Agent. Multi-Ag. 8, 3 (October 2004), 203-236
5. Budinsky, F., Steinberg, D., Ellersick, R., Merks, E., Brodsky, S.A., Grose, T.J.: Eclipse Modeling Framework. Addison Wesley (2003).
6. Ciancarini P. and Wooldridge M. (eds.) Agent-oriented software engineering. Springer Verlag, Berlin, Germany, 2001.
7. Czamecki K. and Helsen. Feature-based survey of model transformation approaches. IBM Systems Journal , 45(3) , 2006
8. EMF and GMF – Basic Tutorial
<http://heim.ifi.uio.no/oddcdb/inf5120//INF5120Tutorial.zip>
(Προσπελάσθηκε 3/11/2009)
9. EuGENia GMF Tutorial , <http://www.eclipse.org/gmt/epsilon/doc/articles/eugenia-gmf-tutorial/> (Προσπελάσθηκε 20/1/2010)
10. Garcia, J., Laguna, M., Carvajal, Y. and Baixauli B.: Requirements variability Support through MDD and graph transformation. In International Workshop on Graph and Model Transformation, pp. 161--173. Tallinn, Estonia (2006)



11. Garcia-Magarino, Ivan, Gomez-Sanz, Jorge J. and Fuentes-Fernandez, Ruben.
2009.The 10th International Workshop on Agent-Oriented Software
Engineering AOSE'09 May 11, 2009, Budapest Hungary
12. Gerber, A., Raymond, K.: Mof to emf: there and back again. In: eclipse '03:
Proceedings of the 2003 OOPSLA workshop on eclipse technology eXchange,
New York, NY, USA, ACM Press (2003) 60-64
13. Giorgini Paolo, Kolp Manuel, Mylopoulos John, Castro Jaelson. Tropos: A
Requirements-Driven Methodology for Agent-Oriented Software. In
Henderson-Sellers B. and Giorgini P. (eds) Agent-Oriented Methodologies.
IdeaGroup Publishing. 2005, pp. 20-45
14. GMF Tooling Tutorial :Step1 ,
<http://127.0.0.1:2711/help/topic/org.eclipse.gmf.doc/tutorials/tooling/step1.html>
(Προσπελάθηκε 15/11/2009)
15. Henderson-Sellers B. and Giorgini P. 2005 Agent-Oriented Methodologies. Idea
Group Publishing.
16. Jeff McAffer and Jean-Michel Lemieux, Eclipse Rich Client Platform: Designing,
Coding, and Packaging Java™ Applications. Addison Wesley, 2005.
17. Jouault F., Bezivin J., KM3: A DSL for Metamodel Specification. In Formal
Methods for Open Object-Based Distributed Systems, Proceedings of the 8th
IFIPWG 6.1 International Conference, (FMOODS 2006), Bologna, Italy, June
14-16, 2006. pp. 171-185.
18. Jouault, F. and Kurtev, I. On the Architectural Alignment of ATL and QVT. In
Proceedings of the 2006 ACM Symposium on Applied Computing (Dijon,
France, April 23-27, 2006). SAC 06. ACM Press, 1188-1195 (2006a)
19. Jouault, F. and Kurtev, I.: Transforming models with ATL. In: Satellite Events at
the MoDELS 2005 Conference. Volume 3844 of Lecture Notes in Computer
Science., Springer-Verlag 128–138 (2006b)
20. Kleppe, A., Warmer, S. and Bast, W. 2003 MDA Explained. The Model Driven
Architecture: Practice and Promise. Addison-Wesley



21. Langlois, B., Jitia, C.E., Jouenne, E.. DSL Classification. In 7th OOPSLA Workshop On Domain-Specific Modeling, 2007.
http://www.dsmforum.org/events/DSM07/papers/langlois_jitia.pdf
(Προσπελάσθηκε 5/2/2010)
22. Matsatsinis, N., Moraitis, P., Psomataki, V. and Spanoudakis, N. "An Agent-Based System for Products Penetration Strategy Selection", Applied Artificial Intelligence Journal, Taylor & Francis, Vol. 17, No. 10, 2003, pp. 901-925.
23. Microsoft Developer Network, Microsoft patterns & practices Developer Center,
<http://msdn.microsoft.com/en-us/library>, (Προσπελάσθηκε 25/4/2010).
24. Moraitis, P., Petraki, E. and Spanoudakis, N.. An Agent-based System for Infomobility Services. In: Proceedings of the third European Workshop on Multi-Agent Systems (EUMAS2005), Brussels, Belgium, December 7 - 8, 2005
25. Moraitis, P., Spanoudakis N.: Argumentation-based Agent Interaction in an Ambient Intelligence Context. IEEE Intelligent Systems, 22(6), pp. 84-93 (2007)
26. Moraitis, P., Petraki, E. and Spanoudakis, N.. Providing Advanced, Personalised Infomobility Services Using Agent Technology. In: Proc. of the 23rd SGAI 191 International Conference on Innovative Techniques and Applications of Artificial Intelligence (AI2003), Peterhouse College, Cambridge, UK, December 15-17, 2003b
27. Moore, S.A. "On conversation policies and the need for exceptions", Issues in Agent Communication, Lecture Notes in Artificial Intelligence 1916, Springer, 2000, pp. 144–159
28. Object and Reference Model Subcommittee (ORMSC) of the OMG Architecture Board: A Proposal for an MDA Foundation Model, white paper
OMGORMSC/ 05-08-01, <http://www.omg.org/cgi-bin/doc?ormsc>
(Προσπελάσθηκε 10/4/2010 –κατόπιν αδείας).



29. Pavon J., and Gomez-Sanz J. (2003). Agent-Oriented Software Engineering with INGENIAS. In: Multi-Agent Systems and Applications III, 3rd International Central and Eastern European Conference on Multi-Agent Systems (CEEMAS'03), Lecture Notes in Computer Science 2691, Springer Verlag, pp.394-403.
30. Pavon, J., Gomez-Sanz, J.J. & Fuentes, R.: The INGENIAS Methodology and Tools. In: Henderson-Sellers, B. and Giorgini, P., editors: Agent-Oriented Methodologies. Idea Group Publishing, pp. 236-276 (2005).
31. Sendall, S. and Kozaczynski, W. 2003. Model Transformation: The Heart and Soul of Model-Driven Software Development. IEEE Softw.20,5(Sep. 2003),42- 45.
32. Spanoudakis, N. and Moraitis, P.. An Ambient Intelligence Application Integrating Agent and Service-Oriented Technologies. In: Proceedings of the 27th SGAI International Conference on Artificial Intelligence (AI2007), Peterhouse College, Cambridge, UK, December 10-12, 2007a
33. Spanoudakis N., Moraitis, P.: Automated Product Pricing Using Argumentation. In: Proceedings of the 5th IFIP Conference on Artificial Intelligence Applications & Innovations (AIAI 2009) Thessaloniki, Greece, April 23-25, 2009.
34. Spanoudakis N., Moraitis, P.: The Agent Systems Methodology (ASEME): A Preliminary Report. In: Proceedings of the 5th European Workshop on Multi-Agent Systems (EUMAS'07), Hammamet, Tunisia, December 13 - 14, 2007b
35. Spaonoudakis, N. The Agent Systems Engineering Methodology (ASEME) , These de Doctorat , Universite Paris Descartes , October 2009.
36. Susi, A., Perini, A., Giorgini, P. and Mylopoulos, J.. The Tropos Metamodel and its Use. Informatica, 29(4):401–408, 2005.
37. The Eclipse Foundation , Platform Plug-in Developer Guide , <http://eclipse.org>, (Προσπελάθηκε 22/4/2010).
38. Weiss, Michael. Patterns for Motivating an Agent-Based Approach. In G. Goos,



J. Hartmanis, and J. van Leeuwen (eds) Conceptual Modeling for Novel
Application Domains, Lecture Notes in Computer Science, Volume
2814/2003, Springer Berlin / Heidelberg, 2003.

39. Wooldridge, M. & Jennings, N. (1995), "Intelligent Agents: Theory and Practice",
The Knowledge Engineering Review 10 (2), 115-152.
40. Wooldridge, M., Jennings, N.R., Kinny, D.: The Gaia Methodology for Agent-
Oriented Analysis and Design. Journal of Autonomous Agents and Multi-
Agent Systems Vol. 3. No. 3 (2000) 285-312.
41. W3schools.com, XML, <http://www.w3schools.com/xml>. (Προσπελάθηκε
12/3/2010).
42. Σκορδαλάκης, Ε., Λογισμική Μηχανική, Εθνικό Μετσόβιο Πολυτεχνείο, 2005.



Παράρτημα Α: Η πλατφόρμα Eclipse

Το Eclipse είναι μια κοινότητα λογισμικού ανοιχτού κώδικα, τα έργα της οποίας εστιάζουν στην κατασκευή μιας ανοιχτής πλατφόρμας ανάπτυξης, η οποία αποτελείται από επεκτάσιμα πλαίσια, εργαλεία και περιβάλλοντα χρόνου εκτέλεσης για την κατασκευή και τη διαχείριση λογισμικού κατά τη διάρκεια ολόκληρου του κύκλου ζωής του. Το ίδρυμα Eclipse είναι ένας μη κερδοσκοπικός οργανισμός που υποστηρίζεται από τα μέλη του και φιλοξενεί τα έργα Eclipse.

Το έργο Eclipse δημιουργήθηκε αρχικά από την IBM το Νοέμβριο του 2001 και υποστηρίζεται από μια κοινοπραξία εταιρειών πώλησης λογισμικού.

Αρχιτεκτονική της πλατφόρμας

Το Eclipse είναι μια πλατφόρμα που έχει σχεδιαστεί από την αρχή για να χτίζονται πάνω της ολοκληρωμένα εργαλεία ανάπτυξης εφαρμογών διαδικτύου και εφαρμογών λογισμικού. Από το σχεδιασμό της, η πλατφόρμα δεν παρέχει μεγάλο επίπεδο χρηστικότητας από μόνη της. Η αξία της όμως βρίσκεται στο γεγονός ότι ενθαρρύνει την ανάπτυξη επιπλέον δυνατοτήτων βασισμένων σε ένα μοντέλο από plug-ins.

Το eclipse προσφέρει μια κοινή διπροσωπία προς το χρήστη (User Interface) για να δουλεύει με εργαλεία. Είναι σχεδιασμένο να λειτουργεί σε πολλά διαφορετικά λειτουργικά συστήματα ενώ ταυτόχρονα παρέχει στιβαρή ολοκλήρωση με το κάθε λειτουργικό σύστημα που βρίσκεται κάτω από αυτό. Τα plug-ins μπορούν να στηρίζονται στα φορητά API's του eclipse και έτσι να λειτουργούν αμετάβλητα σε οποιοδήποτε από τα υποστηριζόμενα λειτουργικά συστήματα.

Στον πυρήνα του eclipse βρίσκεται μια αρχιτεκτονική για δυναμική αναζήτηση, φόρτωση και εκτέλεση των plug-ins. Η πλατφόρμα χειρίζεται τις λειτουργίες της εύρεσης και εκτέλεσης του σωστού κώδικα. Η διπροσωπία χρήστη της πλατφόρμας παρέχει ένα τυποποιημένο μοντέλο πλοήγησης. Κάθε plug-in μπορεί στη συνέχεια να εστιάσει στην σωστή υλοποίηση ενός μικρού αριθμού εργασιών. Παραδείγματα τέτοιων εργασιών είναι η κωδικοποίηση, ο έλεγχος, η μεταγλώττιση, η εκσφαλμάτωση, η δημιουργία διαγραμμάτων, κτλ. [37].



Ανοιχτή αρχιτεκτονική

Η πλατφόρμα eclipse ορίζει μια ανοιχτή αρχιτεκτονική έτσι ώστε η κάθε ομάδα ανάπτυξης plug-in να μπορεί να εστιάζει στη δική της περιοχή εξειδίκευσης. Η ιδέα είναι ότι οι ειδικοί στον τομέα τους χτίζουν το οπίσθιο μέρος και οι ειδικοί στην χρηστικότητα χτίζουν τα εργαλεία που θα χρησιμοποιεί ο τελικός χρήστης (τα plug-ins). Αν η πλατφόρμα είναι καλά σχεδιασμένη, τότε σημαντικές νέες δυνατότητες και νέα επίπεδα ολοκλήρωσης μπορούν να προστεθούν σε αυτήν χωρίς να επηρεάζονται τα υπόλοιπα εργαλεία. Η πλατφόρμα eclipse χρησιμοποιεί το μοντέλο του πάγκου εργασίας (workbench) για να ενσωματώνει τα εργαλεία από τη σκοπιά του τελικού χρήστη. Εργαλεία που αναπτύσσει ο καθένας μπορούν να ενσωματωθούν στον πάγκο εργασίας με τη χρήση καλά ορισμένων διεπαφών που ονομάζονται σημεία επέκτασης (extension points).

Και η ίδια η πλατφόρμα είναι χτισμένη σε στρώματα από plug-ins, καθένα από τα οποία ορίζει επεκτάσεις πάνω σε σημεία επεκτάσεων των plug-ins του χαμηλότερου στρώματος. Με τη σειρά τους και αυτά τα plug-ins ορίζουν τα δικά τους σημεία επέκτασης για να παρέχουν τη δυνατότητα περαιτέρω προσαρμογής από plug-ins ανώτερων στρωμάτων. Αυτό το μοντέλο επέκτασης επιτρέπει σε ομάδες ανάπτυξης plug-ins να προσθέσουν πολλές και διάφορες λειτουργίες στη βασική πλατφόρμα εργαλείων. Τα διάφορα τεχνήματα για κάθε εργαλείο, όπως αρχεία και άλλα δεδομένα, συντονίζονται από ένα κοινό για όλη την πλατφόρμα μοντέλο διαχείρισής τους.

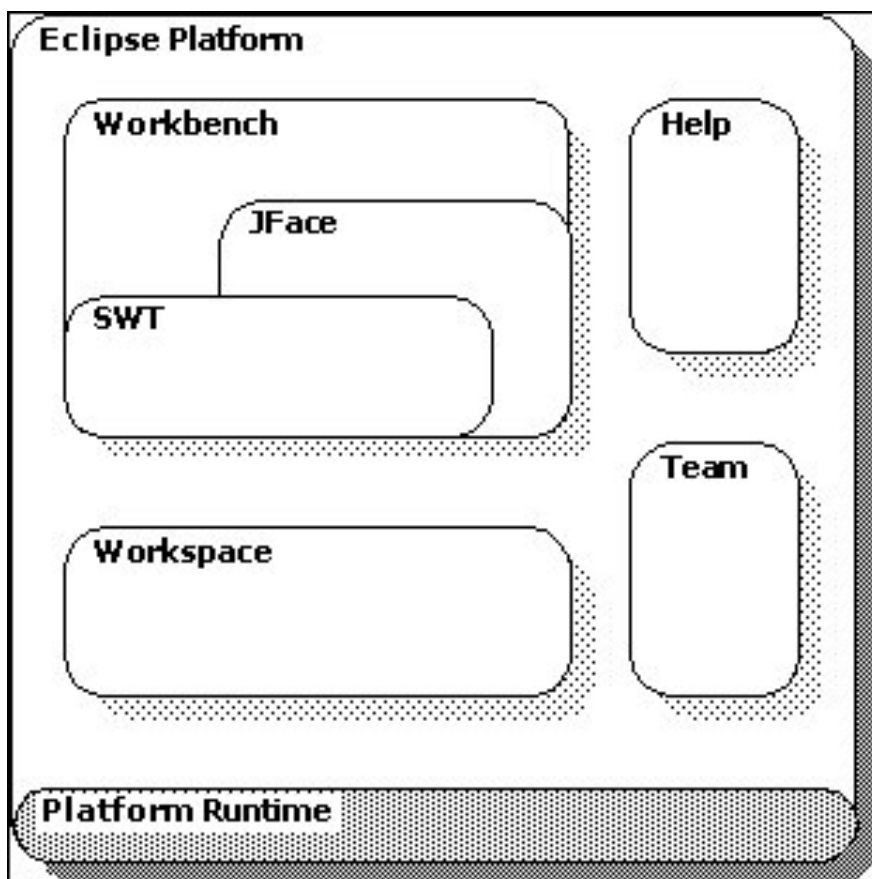
Η πλατφόρμα δίνει στους χρήστες έναν κοινό τρόπο να δουλεύουν με τα εργαλεία και παρέχει διαχείριση των πόρων (resources) που αυτοί δημιουργούν με τα plug-ins. Οι ομάδες ανάπτυξης των plug-ins επωφελούνται επίσης από αυτή την αρχιτεκτονική, καθώς η πλατφόρμα διαχειρίζεται την περιπλοκότητα των διαφορετικών περιβαλλόντων εκτέλεσης, για παράδειγμα των διαφορετικών λειτουργικών συστημάτων.

Η δομή της πλατφόρμας

Η πλατφόρμα είναι δομημένη σαν υποσυστήματα που υλοποιούνται από ένα ή περισσότερα plug-ins το καθένα. Τα υποσυστήματα είναι χτισμένα πάνω σε μια μικρή μηχανή χρόνου εκτέλεσης (runtime engine). Η εικ. 69 δείχνει μια απλοποιημένη όψη. Τα plug-ins που απαρτίζουν ένα υποσύστημα ορίζουν σημεία επέκτασης για πρόσθεση νέας



συμπεριφοράς στο σύστημα.Ο πίνακας 1 περιγράφει τις κύριες ψηφίδες της πλατφόρμας που είναι υλοποιημένες σαν ένα ή περισσότερα plug-ins.



Εικ. 69. Δομή της πλατφόρμας eclipse

Αρχική μορφή της πλατφόρμας

Η βασική πλατφόρμα που παρέχεται στο χρήστη αρχικά είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment, IDE) για οτιδήποτε και τίποτα συγκεκριμένο (εικ. 70).

Τα plug-ins είναι αυτά που καθορίζουν την τελική λειτουργικότητα της πλατφόρμας. Τα νέα plug-ins μπορούν να παρέχουν υποστήριξη για επεξεργασία και διαχείριση επιπρόσθετων τύπων πόρων, όπως αρχεία Java, προγράμματα C, σελίδες HTML και αρχεία JSP.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 70. Αρχική μορφή του πάγκου εργασίας του eclipse

Platform Runtime	Ορίζει το μοντέλο των σημείων επέκτασης και το μοντέλο των plugins. Ανακαλύπτει δυναμικά τα plug-ins και διατηρεί τις πληροφορίες για αυτά και για τα σημεία επέκτασης σε ένα μητρώο (registry) της πλατφόρμας. Τα plug-ins εκκινούνται όταν χρειάζεται, σύμφωνα με τις ενέργειες του χρήστη. Το περιβάλλον χρόνου εκτέλεσης υλοποιείται σύμφωνα με την τεχνολογία OSGi.
------------------	--



Resource management (workspace)	Ορίζει API για δημιουργία και διαχείριση των πόρων (resources), όπως έργα, αρχεία και φάκελοι, οι οποίοι παράγονται από τα εργαλεία και αποθηκεύονται στο σύστημα αρχείων.
Workbench UI	Υλοποιεί τη “θέση οδήγησης” του χρήστη για να περιηγείται μέσα στην πλατφόρμα. Ορίζει σημεία επέκτασης για προσθήκη ψηφίδων διπροσωπίας χρήστη όπως όψεις ή ενέργειες μενού. Παρέχει πρόσθετες εργαλειοθήκες (JFace και SWT) για δημιουργία διεπαφών χρήστη. Οι υπηρεσίες διπροσωπίας χρήστη είναι δομημένες έτσι ώστε ένα υποσύνολο των plug-ins μπορεί να χρησιμοποιηθεί για να δημιουργηθούν εφαρμογές rich client οι οποίες να είναι ανεξάρτητες από το μοντέλο του workspace. Τα plug-ins που σχετίζονται με εφαρμογές ολοκληρωμένου περιβάλλοντος ανάπτυξης (Integrated Development Environment, IDE) ορίζουν επιπλέον τρόπους πλοήγησης και διαχείρισης των πόρων
Help system	Ορίζει σημεία επέκτασης που τα plug-ins μπορούν να χρησιμοποιούν για να παρέχουν βοήθεια ή άλλου είδους τεκμηρίωση στο χρήστη.
Team support	Ορίζει ένα μοντέλο προγραμματισμού ομάδας, για τη διαχείριση και την προσθήκη αριθμών έκδοσης στους πόρους.
Debug support	Ορίζει ένα μοντέλο εκσφαλμάτωσης και κλάσεις διπροσωπίας χρήστη για το χτίσιμο εκσφαλματωτών. Το μοντέλο και οι κλάσεις αυτές είναι ανεξάρτητες από τη γλώσσα προγραμματισμού.
Other utilities	Άλλα plug-ins παρέχουν διάφορες άλλες χρήσιμες λειτουργίες όπως αναζήτηση και σύγκριση πόρων, μεταγλώττιση με χρήση αρχείων ρυθμίσεων σε γλώσσα XML, και δυναμική ανανέωση



	της πλατφόρμας από έναν server.
--	---------------------------------

Πίνακας 1: Περιγραφή των ψηφίδων της πλατφόρμας του eclipse

Επέκταση της πλατφόρμας με plug-ins

Όπως αναφέρθηκε και παραπάνω, η πλατφόρμα eclipse είναι δομημένη ως εξής: στον πυρήνα υπάρχει μια μηχανή χρόνου εκτέλεσης (core runtime engine) και πάνω της υπάρχει ένα σύνολο επιπρόσθετων λειτουργιών που εγκαθίστανται σαν plug-ins. Τα plug-ins συνεισφέρουν λειτουργικότητα στην πλατφόρμα συνεισφέροντας σε προκαθορισμένα σημεία επέκτασης. Ένα παράδειγμα ενός τέτοιου plug-in είναι και ο πάγκος εργασίας. Η γλώσσα στην οποία είναι υλοποιημένο το Eclipse είναι η Java. Όταν όμως εκκινείται ο πάγκος εργασίας, δεν εκκινείται ένα απλό πρόγραμμα Java. Ενεργοποιείται η μηχανή εκτέλεσης, η οποία μπορεί δυναμικά να ανακαλύπτει εγκατεστημένα plug-ins και να τα εκκινεί όποτε χρειάζεται.

Από τα παραπάνω προκύπτει ότι όταν κάποιος θέλει να γράψει κώδικα ο οποίος επεκτείνει την πλατφόρμα, η διαδικασία που πρέπει να ακολουθήσει είναι διαφορετική από τη διαδικασία ανάπτυξης μιας απλής εφαρμογής Java. Συγκεκριμένα, θα πρέπει να ορίσει επεκτάσεις μέσα στο plug-in του, οι οποίες να συνδέονται σε κάποια από τα σημεία επεκτάσεων που έχει ορίσει η πλατφόρμα. Από τη σκοπιά της πλατφόρμας, το plug-in που γράφει κάποιος τρίτος δε διαφέρει από τα βασικά plug-ins σαν αυτό που χειρίζεται τους πόρους (resources) ή τον ίδιο τον πάγκο εργασίας. Τα βήματα που πρέπει να ακολουθήσουμε είναι:

- Απόφαση για τον τρόπο με τον οποίο το plug-in θα ενσωματωθεί στην πλατφόρμα.
- Εύρεση των κατάλληλων σημείων επέκτασης στα οποία χρειάζεται να συνεισφέρουμε ώστε να ενσωματωθεί το plug-in.
- Υλοποίηση αυτών των επεκτάσεων σύμφωνα με την προδιαγραφή για τα σημεία επέκτασης.
- Δημιουργία ενός αρχείου manifest (MANIFEST.MF) το οποίο περιγράφει το πακέταρισμα και τα προαπαιτούμενα για τον κώδικά μας, και ενός plug-in manifest (plugin.xml) το οποίο περιγράφει τις επεκτάσεις που ορίζουμε.



Ένα ελάχιστο plug-in

Το πιο απλό πρόγραμμα γραμμένο σε γλώσσα Java χωρίς τη χρήση άλλων βιβλιοθηκών διαπροσωπίας χρήστη μοιάζει με τον κώδικα 1.

Πηγαίος Κώδικας 1: Το πρόγραμμα Hello World σε απλή Java

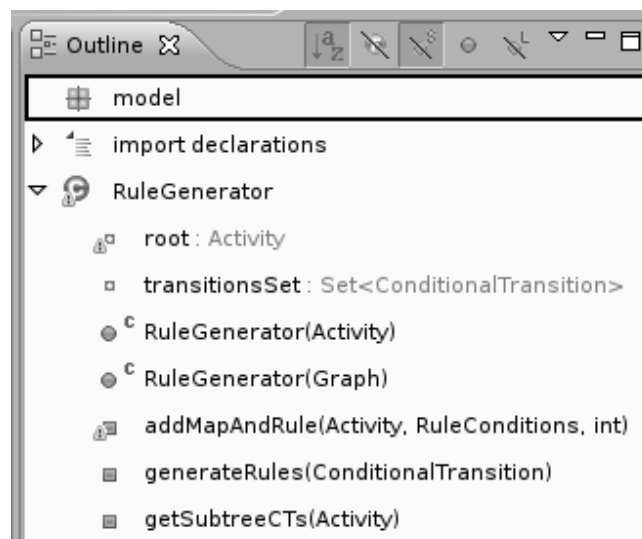
```
public class HelloWorld {  
    public static void main (String[] args) {  
        System.out.println("Hello World");  
    }  
}
```

Στο πλαίσιο της πλατφόρμας Eclipse, αυτό το πρότυπο μεταβάλλεται. Αντί να το σκεφτόμαστε ως ένα ανεξάρτητο πρόγραμμα, το αντιμετωπίζουμε ως μια επέκταση της πλατφόρμας. Γενικά, υπάρχουν αρκετοί διαφορετικοί τρόποι με τους οποίους μπορεί να επεκτείνει κανείς το περιβάλλον χρήστη του πάγκου εργασίας. Ο πάγκος εργασίας είναι ένα παράθυρο που αποτελείται από διάφορα οπτικά μέρη (αντικείμενα). Αυτά τα μέρη χωρίζονται σε δύο βασικές κατηγορίες: τις όψεις (views) και τους επεξεργαστές (editors).

Οι επεξεργαστές επιτρέπουν στο χρήστη να επεξεργάζεται κάτι μέσα στον πάγκο εργασίας. Οι επεξεργαστές είναι “εγγραφο–κεντρικοί” και μοιάζουν σε πολλά με τους γνωστούς επεξεργαστές αρχείων (π.χ. αρχείων κειμένου). Τα έγγραφα ακολουθούν έναν κύκλο ζωής άνοιγμα – αποθήκευση – κλείσιμο. Οι επεξεργαστές είναι στενά δεμένοι και ενσωματωμένοι στον πάγκο εργασίας. Ο πάγκος εργασίας προσφέρει έναν έτοιμο editor για αρχεία απλού κειμένου. Παρέχει επίσης μια κοινή διαπροσωπεία με το χρήστη για τις συνηθισμένες εργασίες (π.χ. κουμπιά για άνοιγμα, αποθήκευση, αποθήκευση ως, κλπ.), αλλά ο τρόπος με τον οποίο γίνονται αυτές οι εργασίες υλοποιείται ξεχωριστά από κάθε editor. Οι όψεις παρέχουν πληροφορίες σχετικά με κάποιο αντικείμενο με το οποίο εργάζεται ο χρήστης στον πάγκο εργασίας. Συχνά οι όψεις αλλάζουν τα περιεχόμενά τους όταν ο χρήστης επιλέγει διαφορετικά αντικείμενα στον πάγκο εργασίας. Πολλές φορές οι όψεις δρουν συμπληρωματικά με τους επεξεργαστές, παρέχοντας πληροφορίες σχετικά με το περιεχόμενο τους. Είναι δυνατό μία όψη να είναι αρκετά γενική ώστε να μπορεί να χρησιμοποιείται σε πολλές διαφορετικές περιστάσεις με διαφορετικούς τρόπους. Για παράδειγμα, η όψη ιδιοτήτων (Properties View) δείχνει πληροφορίες όπως πλήρες



όνομα, κατάλογος, μέγεθος, ημερομηνία τελευταίας τροποποίησης, κτλ. όταν ο χρήστης επιλέξει κάποιο αρχείο μέσα στην όψη Navigator (βλ. εικ. 70), αλλά μέσα σε έναν επεξεργαστή αρχείων XML μπορεί να δείχνει ιδιότητες όπως όνομα, τύπο, κτλ. του επιλεγμένου στοιχείου ή ιδιότητας XML. Ομοίως, η όψη περίληψης (Outline View, εικ. 71), όταν ο χρήστης κάνει επεξεργασία πηγαίου κώδικα δείχνει μια σύνοψη του ενεργού αρχείου που αποτελείται από δηλώσεις, ονόματα και τύπους μεταβλητών και μεθόδων, η εμβέλεια των οποίων επισημαίνεται με διαφορετικά χρώματα/σχήματα, αλλά μπορεί να χρησιμοποιηθεί και αλλιώς.



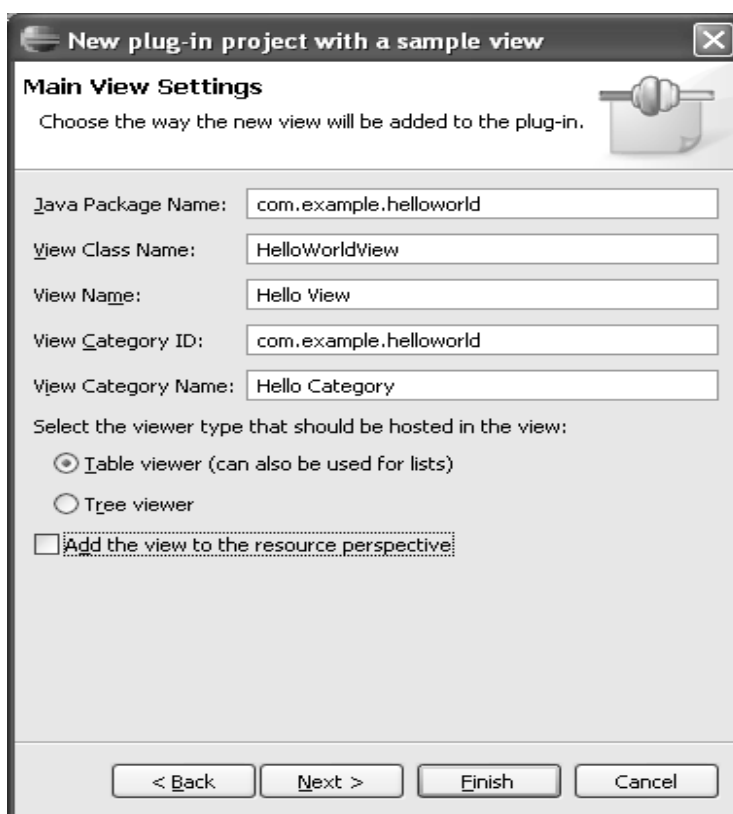
Εικ. 71. Όψη Outline για ένα αρχείο πηγαίου κώδικα Java

Δημιουργία του project για το plug-in

Για να δημιουργήσουμε ένα plug-in για την πλατφόρμα Eclipse μπορούμε να χρησιμοποιήσουμε οποιοδήποτε Java IDE, αλλά το ίδιο το Eclipse SDK προσφέρει επιπλέον εργαλεία για την ανάπτυξη plug-ins. Συγκεκριμένα, υπάρχουν οι ευκολίες γεννήτορα κώδικα του Plug-in Development Environment (PDE) που μας δίνουν ένα template από το οποίο μπορούμε να ξεκινήσουμε. Το PDE αρχικοποιεί ένα project στο οποίο μπορούμε να γράψουμε κώδικα Java και δημιουργεί τα προκαθορισμένα αρχεία manifest του plug-in (τα οποία θα εξηγηθούν αναλυτικότερα παρακάτω) καθώς και μια κλάση η οποία υλοποιεί την όψη μας.



Η δημιουργία ενός νέου project για ένα plug-in είναι μια διαδικασία που μπορεί να γίνει μέσα στο Eclipse SDK με τη βοήθεια ενός wizard (βλ. εικ. 72) . Η διαδικασία είναι απλή και δεν αξίζει να αναφερθεί πιο αναλυτικά εδώ. Το αποτέλεσμα της είναι ένα project δομημένο ώστε να περιέχει αρκετούς φακέλους, αρχεία και ένα πακέτο Java. Τα σημαντικά αρχεία σε αυτό το στάδιο είναι το plugin.xml και το MANIFEST.MF (αρχεία manifest) καθώς και ο κώδικας Java για το plug-in.



Εικ. 72. Δημιουργία ενός νέου plug-in με τη βοήθεια του PDE

Η όψη Hello World

Για να μετατρέψουμε το παραπάνω πρόγραμμα Hello World σε plug-in του Eclipse μπορούμε να υλοποιήσουμε μια νέα όψη (view) η οποία να εμφανίζει στο χρήστη το μήνυμα “Hello World”. Το plug-in org . eclipse . ui .workbench ορίζει τις περισσότερες από τις δημόσιες διεπαφές (interfaces) που απαρτίζουν το API του πάγκου εργασίας. Αυτές οι διεπαφές βρίσκονται στο πακέτο org . eclipse . ui και στα υποπακέτα του.



Πολλές από αυτές τις διεπαφές έχουν προκαθορισμένες κλάσεις υλοποίησης, τις οποίες μπορούμε να επεκτείνουμε για να παρέχουμε απλές τροποποιήσεις στο σύστημα.

Ο κώδικας 2 περιέχει την υλοποίηση σε γλώσσα Java της όψης Hello World.

Πηγαίος Κώδικας 2: Η όψη Hello World

```
package com . example . hello world ;  
import org . eclipse . swt . widgets . Composite ;  
import org . eclipse . swt . widgets . Label ;  
import org . eclipse . swt . SWT ;  
import org . eclipse . ui . part . View Part ;  
public class HelloWorldView extends View Part {  
    Label label ;  
    public HelloWorldView ( ) {  
    }  
    public void createPartControl ( Composite parent ) {  
        label = new Label ( parent , SWT . WRAP ) ;  
        label . setText ( "Hello World" ) ;  
    }  
    public void setFocus ( ) {  
    }  
}
```

Αρχεία manifest

Όπως αναφέραμε, το PDE αναλαμβάνει και την αρχικοποίηση δύο αρχείων manifest που περιγράφουν κάθε plug-in που δημιουργεί.

Το plugin.xml (κώδικας 3) συμπεριλαμβάνει όλες τις πληροφορίες σχετικά με την επέκτασή μας και το πώς αυτή εκτελείται μέσα σε μια συσκευασία XML. Στη συγκεκριμένη περίπτωση δηλώνεται, μέσα στο στοιχείο extension, ότι το νέο plug-in επεκτείνει την πλατφόρμα στο σημείο επέκτασης. Το SWT (Standard Widget Toolkit) είναι μια εργαλειοθήκη γραφικών η οποία μπορεί να χρησιμοποιηθεί με την πλατφόρμα Java. Αναπτύχθηκε αρχικά από την IBM και τώρα συντηρείται από το ίδρυμα Eclipse, όπως και το Eclipse IDE.



Είναι μια εναλλακτική λύση των εργαλειοθηκών γραφικής διαπροσωπείας χρήση AWT και Swing, τις οποίες παρέχει η Sun Microsystems μαζί με την πλατφόρμα Java.

Το SWT είναι γραμμένο σε γλώσσα Java. Για να εμφανίζει τα γραφικά στοιχεία, χρησιμοποιεί διαφορετικές, τοπικές (native) γραφικές βιβλιοθήκες για κάθε λειτουργικό σύστημα (π.χ. Motif ή GTK+ για λειτουργικά συστήματα της οικογένειας Unix, Win32 για Microsoft Windows, Carbon ή Cocoa για Mac OS), με αποτέλεσμα "η εμφάνιση και αίσθηση" μιας γραφικής εφαρμογής που βασίζεται στο SWT να ταιριάζει με τις υπόλοιπες εφαρμογές του λειτουργικού συστήματος.

Αυτό έρχεται σε αντίθεση με τις εφαρμογές γραμμένες σε AWT και Swing, οι οποίες έχουν μια ενιαία εμφάνιση σε όλα τα λειτουργικά συστήματα.

Η πλατφόρμα Eclipse και όλα τα plug-ins που είναι γραμμένα για αυτήν χρησιμοποιούν το SWT ως εργαλειοθήκη γραφικής διαπροσωπείας χρήση.

org.eclipse.ui.views. Το αναγνωριστικό (id) της επέκτασης, δηλαδή το όνομα με το οποίο η πλατφόρμα γνωρίζει το plug-in είναι το example.HelloWorldView και η κλάση Java που υλοποιεί την επέκταση είναι η com.example.helloworld.HelloWorldView. Το plugin ορίζει επίσης και μια νέα κατηγορία όψεων, τη Hello Category.

Πηγαίος Κώδικας 3: plugin.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?eclipse version="3.0"?>
<plugin>
<extension point="org.eclipse.ui.views">
<category
name="Hello Category"
id="example.helloworld">
</category>
<view
name="Hello View"
icon="icons/sample.gif"
category="com.example.helloworld"
class="com.example.helloworld.HelloWorldView">
```



```
id="example.HelloWorldView">
```

```
</view>
```

```
</extension>
```

```
</plugin>
```

Το άλλο αρχείο που δημιουργείται αυτοματα είναι το MANIFEST.MF (κώδικας 4). Το αρχείο αυτό περιγράφει πληροφορίες χαμηλότερου επιπέδου σχετικά με το πακετάρισμα του plug-in, χρησιμοποιώντας ορολογία της τεχνολογίας OSGi. Περιέχει πληροφορίες όπως το όνομα του plug-in (bundle), τα bundles τα οποία απαιτεί να υπάρχουν, την ελάχιστη έκδοση του eclipse, κλπ.

Πηγαίος Κώδικας 4: MANIFEST.MF

```
Manifest-Version : 1.0
```

```
Bundle-ManifestVersion : 2
```

```
Bundle-Name: HelloWorld Plug-in
```

```
Bundle-SymbolicName : com.example.helloworld;singleton:=true
```

```
Bundle-Version : 1.0.0
```

```
Require-Bundle : org.eclipse.ui,
```

```
org.eclipse.ui.views
```

Και τα δύο αρχεία manifest μπορούμε να τα επεξεργαζόμαστε είτε απευθείας είτε μέσω των γραφικών ευκολιών που προσφέρει το PDE και συγκεκριμένα ο plug-in editor.

Εκτέλεση του plug-in

Μετά τη μεταγλώττιση της νέας όψης, υπάρχουν δύο διαφορετικοί πιθανοί τρόποι, με τους οποίους μπορεί να εκτελεστεί.

- Τοποθετούμε τα αρχεία manifest και το αρχείο jar στον κατάλογο eclipse/plugins. Όταν ο πάγκος εργασίας επανεκκινηθεί, θα βρει το νέο plug-in.
- Το εργαλείο PDE μπορεί να χρησιμοποιηθεί για να εκτελεστεί ένας άλλος πάγκος εργασίας μέσα από τον τρέχοντα πάγκο. Αυτή η μέθοδος χρησιμοποιείται κατά τη διάρκεια της ανάπτυξης του plug-in και είναι πιο βολική για τον προγραμματιστή γιατί μπορεί να δοκιμάζει τα νέα plugins που δημιουργεί άμεσα.



Όπως και αν επιλέξουμε να εκτελέσουμε το plug-in Hello World, η όψη θα μπορεί να εμφανιστεί από το μενού Window > Show View > Hello Category > Hello View του πάγκου εργασίας του eclipse (δεδομένου ότι το plug-in παρέχει μια επέκταση για το σημείο επέκτασης org.eclipse.views, όπως ορίστηκε στο plugin.xml).

Μέχρι τη στιγμή που θα επιλέξει ο χρήστης να εμφανίσει την όψη Hello View, ο κώδικας του plug-in δεν έχει εκτελεστεί. Η δήλωση μέσα στο plugin.xml είναι αρκετή για να γνωρίζει ο πάγκος εργασίας ότι υπάρχει μια όψη με αυτό το όνομα στην κατηγορία Hello Category καθώς και την κλάση η οποία υλοποιεί την όψη. Τη στιγμή που θα επιλέξει ο χρήστης να δείξει την όψη ο πάγκος εργασίας ενεργοποιεί το plug-in, αρχικοποιεί την κλάση HelloWorldView και τώρα ο κώδικας του plug-in εκτελείται.

Πλατφόρμα Rich Client (Rich Client Platform)

Γενικά

Ενώ η πλατφόρμα Eclipse είναι σχεδιασμένη να λειτουργεί ως μια πλατφόρμα ανοιχτών εργαλείων, η αρχιτεκτονική της είναι τέτοια που οι ψηφίδες της θα μπορούσαν να χρησιμοποιηθούν για να κατασκευαστεί σχεδόν οποιαδήποτε εφαρμογή πελάτη.

Εφαρμογές διαφορετικές από IDEs μπορούν να κατασκευαστούν με χρήση ενός υποσυνόλου της πλατφόρμας. Αυτές οι “πλούσιες” εφαρμογές εξακολουθούν να βασίζονται σε ένα μοντέλο δυναμικών plug-ins και η διαπροσωπεία χρήστη κατασκευάζεται με τις ίδιες εργαλειοθήκες και τα ίδια σημεία επέκτασης. Η διάταξη και η λειτουργικότητα του πάγκου εργασίας βρίσκονται υπό τον έλεγχο της ομάδας ανάπτυξης του plug-in.

Η πλατφόρμα Rich Client (RCP) είναι το ελάχιστο σύνολο των plug-ins που χρειάζονται για να κατασκευαστεί μια εφαρμογή με διαπροσωπεία χρήστη βασισμένη στην πλατφόρμα. Μια εφαρμογή RCP απαιτεί κατ' αρχήν μόνο δύο plug-ins, τα org . eclipse . ui και org . eclipse . core . runtime, μαζί με τις εξαρτήσεις τους.

Παρόλα αυτά, μια εφαρμογή rich client μπορεί να χρησιμοποιεί επιπλέον οποιαδήποτε άλλη ψηφίδα που χρειάζεται για τη λειτουργικότητά της, και μπορεί να απαιτεί οποιοδήποτε άλλο plug-in εκτός από τα απολύτως απαραίτητα. Παραδείγματα είναι το σύστημα βοήθειας προς το χρήστη και ο διαχειριστής αναβαθμίσεων.



Ιστορία του Eclipse ως πλατφόρμα RCP

Το έργο Eclipse δεν ξεκίνησε με το σκοπό να δημιουργήσει μια πλατφόρμα RCP. Ο αρχικός στόχος ήταν να δημιουργηθεί μια πλατφόρμα για να ενσωματώνονται σε αυτήν εργαλεία ανάπτυξης εφαρμογών, δηλαδή ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE). Το Eclipse σαν RCP ξεκίνησε κατά τη διάρκεια της έκδοσης 2.1, όταν κάποιοι παρατήρησαν ότι τα IDEs που βασίζονταν πάνω στο Eclipse είχαν καλή λειτουργικότητα, εμφάνιση και επιδόσεις, και ότι το ίδιο πλαίσιο που θα μπορούσε να χρησιμοποιηθεί για να χτιστούν εφαρμογές για πιο γενική χρήση.

Η παρατήρηση ήταν ορθή, αλλά η υλοποίηση της ιδέας είχε πολλές πρακτικές δυσκολίες. Οι κυριότερες ήταν όλες οι υποθέσεις που είχαν γίνει σχετικά με το Eclipse ως μια πλατφόρμα δημιουργίας εργαλείων και η συνακόλουθη έλλειψη δυνατότητας αλλαγής κάποιων στοιχείων της εμφάνισης και της λειτουργικότητας της πλατφόρμας.

Η έκδοση 3.0 ήταν ένας σημαντικός σταθμός για το Eclipse ως μια πλατφόρμα RCP. Όλες οι αλληλεξαρτήσεις που σχετίζονταν με τη λειτουργικότητα ως IDE εξαλείφθηκαν και στα διάφορα μέρη της διαπροσωπείας με το χρήστη προστέθηκαν νέες δυνατότητες ρυθμίσεων. Εισήχθη το περιβάλλον χρόνου εκτέλεσης το βασισμένο στο OSGi και έτσι τέθηκαν οι βάσεις για δυναμική εγκατάσταση, διαγραφή και αναβάθμιση των plug-ins. Αυτές οι δύο μεγάλες αλλαγές σήμαιναν μια μεγάλη αναδιάρθρωση της πλατφόρμας.

Μετατροπή ενός απλού plug-in σε εφαρμογή RCP

Το εργαλείο PDE μπορεί εύκολα να χρησιμοποιηθεί για να δημιουργηθεί και μια εφαρμογή RCP, αφού περιέχει μια τέτοια επιλογή στον wizard για δημιουργία νέου plug-in. Οι αλλαγές που χρειάζεται να γίνουν είναι μικρές: στο αρχείο plugin.xml προστίθεται μια επέκταση στο σημείο επέκτασης

org.eclipse.core.runtime.applications (κώδικας 5).

Πηγαίος Κώδικας 5: plugin.xml για εφαρμογή RCP

```
<plugin>  
<extension  
id="application"  
point="org.eclipse.core.runtime.applications">
```




```
<application>  
<run  
class="merged.Application">  
</run>  
</application>  
</extension>  
<!--....-->  
</plugin>
```

Μέσα στο στοιχείο run ορίζεται η κλάση η οποία αναλαμβάνει να ελέγχει όλες τις πλευρές της εκτέλεσης της εφαρμογής, δηλαδή την εκκίνηση, τον τερματισμό και πιθανώς την επανεκκίνησή της. Πρόκειται για μια απλή κλάση που υλοποιεί τη διαπροσωπεία IApplication, δηλαδή τις μεθόδους start () και stop () και μπορεί να μοιάζει με τον κώδικα 6.

Πηγαίος Κώδικας 6: Application.java

```
import org.eclipse.equinox.app.IApplication;  
import org.eclipse.equinox.app.IApplicationContext;  
import org.eclipse.swt.widgets.Display;  
import org.eclipse.ui.IWorkbench;  
import org.eclipse.ui.PlatformUI;  
public class Application implements IApplication {  
    public Object start(IApplicationContext context)  
        throws Exception {  
        Display display = PlatformUI.createDisplay();  
        try {  
            int returnCode = PlatformUI.createAndRunWorkbench  
                (display, new ApplicationWorkbenchAdvisor());  
            if (returnCode == PlatformUI.RETURN_RESTART)  
                return IApplication.EXIT_RESTART;  
            else  
                return IApplication.EXIT_OK;  
        }  
    }  
}
```



```
}finally{  
display.dispose();  
}  
}  
publicvoidstop(){//...  
}
```

Η κλάση Application δημιουργεί και εκτελεί τον πάγκο εργασίας. Τώρα πια το αρχικό plug-in έχει μετατραπεί σε μια πλήρη εφαρμογή που μπορεί να στέκεται μόνη της.

Προϊόντα Eclipse

Ένα προϊόν eclipse είναι μια επέκταση στο σημείο επέκτασης της πλατφόρμας που λέγεται org.eclipse.core.runtime.products. Η μετατροπή μιας εφαρμογής RCP σε ένα προϊόν eclipse, παρουσιάζει κάποια πλεονεκτήματα για έναν προγραμματιστή.

Όταν διανέμεται η εφαρμογή στους τελικούς χρήστες, πρέπει έχουν τοποθετηθεί σε ένα πακέτο όλα τα plug-ins-εξαρτήσεις που χρειάζονται για να εκτελεστεί αυτή, και να διανέμονται και αυτά μαζί με τον αντικειμενικό κώδικα του “κυρίως” plug-in. Σε πρώτο επίπεδο, η πληροφορία για τις άμεσες εξαρτήσεις του plug-in υπάρχει στο αρχείο MANIFEST.MF, αλλά καθένα από τα plug-ins-εξαρτήσεις έχει πιθανόν και δικές του εξαρτήσεις που απαιτεί το ίδιο για να λειτουργήσει, με αποτέλεσμα τη δημιουργία ενός δέντρου εξαρτήσεων. Για ένα προϊόν eclipse, ο υπολογισμός όλων αυτών των εξαρτήσεων μπορεί να γίνεται αυτόματα από το PDE, και στη συνέχεια υποστηρίζεται και η εξαγωγή της εφαρμογής σε μορφή έτοιμη να εκτελεστεί για διάφορα λειτουργικά συστήματα (linux, macosx, solaris, win32).

Επίσης, χρησιμοποιώντας ένα προϊόν μπορεί κανείς να ορίσει και να παραμετροποιήσει την εμφάνιση του Eclipse για μια συγκεκριμένη εφαρμογή.

Στον κώδικα 7 φαίνεται ότι με αυτόν τον τρόπο μπορεί για παράδειγμα να οριστεί μια σύντομη περιγραφή (aboutInfo) με εικόνα και μικρό κείμενο της εφαρμογής. Υπάρχουν και άλλα στοιχεία εμφάνισης που ρυθμίζονται εδώ, όπως τα εικονίδια που θα εμφανίζονται στα διάφορα λειτουργικά συστήματα, ή η εμφάνιση μιας αρχικής οθόνης (splash) την ώρα που εκκινείται η εφαρμογή, τα οποία έχουν παραλειφθεί για συντομία.



Τέλος, φαίνεται η λίστα όλων των plu-ins που απαιτούνται για τη λειτουργία της εφαρμογής.

Η δημιουργία και η επεξεργασία του αρχείου .product είναι απλές διαδικασίες και γίνονται μέσα από το περιβάλλον ανάπτυξης PDE.

Πηγαίος Κώδικας 7: RCPSS.product

```
<?xml version="1.0" encoding="UTF-8"?>
<?pde version="3.4"?>
<product name="RCP Simple Sequencing Editor"
id="merged.product"
application="merged.application"
useFeatures="false">
<aboutInfo>
<image path="/merged/images/about.png"/>
<text>
RCP Simple Sequencing Editor v0.1.0
A Graphical Editor that creates Simple Sequencing Documents
by Christos Gitsis, Andreas Papasalouros
(c) Copyright authors
</text>
</aboutInfo>
<configIni use="default">
</configIni>
<launcherArgs>
<programArgs>-console -noExit</programArgs>
<vmArgsMac>-XstartOnFirstThread
-Dorg.eclipse.swt.internal.carbon.smallFonts
</vmArgsMac>
</launcherArgs>
<!.....>
<plugins>
```



```
<plugin id="com.ibm.icu"/>  
<plugin id="javax.servlet"/>  
<plugin id="org.apache.ant"/>  
<plugin id="org.eclipse.core.commands"/>  
<plugin id="org.eclipse.core.filesystem"/>  
<plugin id="org.eclipse.core.filesystem.linux.x86_64"  
fragment="true"/>  
<! ... σύνολο 59 plugins >  
</plugins>  
</product>
```



Παράρτημα Β : Το πλαίσιο Graphical Editing Framework

Το πλαίσιο Graphical Editing Framework (GEF) παρέχει το υπόβαθρο ώστε μια ομάδα ανάπτυξης λογισμικού να μπορεί να δημιουργήσει έναν πλούσιο, διαδραστικό γραφικό επεξεργαστή (editor) από ένα ήδη υπάρχον μοντέλο. Αρχιτεκτονικά, το GEF αποτελείται από δύο plug-ins:

Draw2d (org.eclipse.draw2d) Μια ελαφριά εργαλειοθήκη γραφικών για σχεδίαση και διάταξη οπτικών αντικειμένων πάνω σε έναν καμβά SWT.

GEF (org.eclipse.gef) ένα διαδραστικό πλαίσιο αρχιτεκτονικής MVC, χτισμένο πάνω στο Draw2d.

Ο όρος “GEF” είναι πιθανό να αναφέρεται σε ολόκληρη την ψηφίδα (2 plug-ins) ή μόνο στο plug-in org . eclipse . gef.

Αρχιτεκτονική MVC

Το αρχιτεκτονικό μόρφημα Model-View-Controller (MVC) [23,42] διαμερίζει μια διαδραστική εφαρμογή σε τρεις συνιστώσες. Το μοντέλο (model) περιέχει τη βασική λειτουργικότητα και τα δεδομένα.

Οι όψεις (views) και οι ελεγκτές (controllers) από κοινού συνιστούν τη διαπροσωπεία χρήστη. Ένας μηχανισμός αλλαγής-ενημέρωσης εξασφαλίζει τη συνέπεια μεταξύ της διαπροσωπείας χρήστη και του μοντέλου. Αναλυτικότερα:

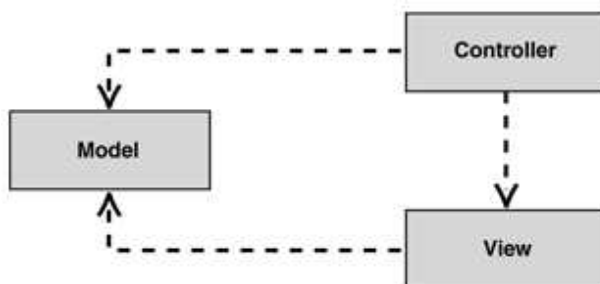
Το μοντέλο είναι η ειδική για το πεδίο με το οποίο ασχολείται η εφαρμογή αναπαράσταση της πληροφορίας, μαζί με τους κανόνες που καθορίζουν τους τρόπους πρόσβασης σε αυτήν και τους τρόπους μεταβολής της. Συνήθως το μοντέλο είναι προσέγγιση μιας διεργασίας από τον πραγματικό κόσμο.

Η όψη μετατρέπει το μοντέλο σε μια μορφή κατάλληλη για διάδραση, συνήθως σε ένα στοιχείο διαπροσωπείας χρήστη. Η όψη έχει την ευθύνη να διατηρεί σε κάθε στιγμή μια συνεπή αναπαράσταση του μοντέλου, όταν αυτό μεταβάλλεται. Είναι δυνατό να υπάρχουν πολλαπλές όψεις για το ίδιο μοντέλο, οι οποίες να εξυπηρετούν διαφορετικούς σκοπούς.

Ο ελεγκτής είναι η συνιστώσα που αντιδρά σε συμβάντα (συνήθως ενέργειες του χρήστη) πάνω στην όψη, τα επεξεργάζεται και τα μεταφράζει σε ενέργειες που θα εκτελεστούν από το μοντέλο.



Αργυρίου Αντώνιος, 'Μετασχηματισμοί μοντέλων : Από την ανάλυση προδιαγραφών στην ανάλυση και την σχεδίαση ενός συστήματος'



Εικ. 73. Αρχιτεκτονικό μόρφημα MVC

Είναι σημαντικό να σημειωθεί ότι τόσο ο ελεγκτής όσο και η όψη έχουν εξάρτηση από το μοντέλο. Αντίθετα, το μοντέλο δεν εξαρτάται από καμία από τις άλλες δύο συνιστώσες, και αυτό είναι ένα από τα σημαντικότερα πλεονεκτήματα της αποσύζευξης των συνιστωσών. Ο διαχωρισμός αυτός επιτρέπει την κατασκευή και τον έλεγχο του μοντέλου ανεξάρτητα από την οπτική αναπαράστασή του.

Draw2d

Γενικά

Το Draw2d είναι μια “ελαφριά” εργαλειοθήκη γραφικών ψηφίδων, οι οποίες ονομάζονται σχήματα (figures). Ο όρος ελαφριά σημαίνει ότι ένα σχήμα είναι απλώς ένα αντικείμενο Java, χωρίς κανέναν πόρο δεσμευμένο στο λειτουργικό σύστημα που να αντιστοιχεί σε αυτό. Τα σχήματα μπορούν να συντεθούν μέσω μιας σχέσης παιδιού-γονέα. Κάθε σχήμα έχει ορθογώνια όρια (bounds), μέσα στα οποία ζωγραφίζονται το ίδιο και τα παιδιά του. Ένας διαχειριστής διάταξης μπορεί να χρησιμοποιηθεί για να τοποθετήσει τα παιδιά ανάλογα με τον αύξοντα αριθμό τους ή/και με τον περιορισμό τους. Ένα ελαφρύ σύστημα (LightweightSystem) συσχετίζει μια σύνθεση σχημάτων με έναν καμβά SWT. Το ελαφρύ σύστημα τοποθετεί ακροατές (listeners) για τα περισσότερα συμβάντα SWT, και προωθεί τα περισσότερα από αυτά σε έναν αποστολέα συμβάντων (EventDispatcher).

Αντίθετα, τα συμβάντα που σχετίζονται με τη σχεδίαση, προωθούνται στον διαχειριστή ανανεώσεων (UpdateManager) ο οποίος συντονίζει τη σχεδίαση και τη διάταξη των σχημάτων. Τα σχήματα μπορούν να ανανεωθούν με τρόπους που επηρεάζουν την εμφάνιση ή το μέγεθός τους, και ο διαχειριστής ανανεώσεων εξασφαλίζει ότι μόνο μια επαναδιάταξη συμβαίνει, ακολουθούμενη από επανασχεδίαση μόνο της περιοχής που



ανανεώθηκε. Κάθε τύπος διαγράμματος, εγγράφου ή σχεδίου μπορεί να κατασκευαστεί εύκολα και να ανανεώνεται αποδοτικά με συνδυασμένη χρήση των ήδη παρεχόμενων από το Draw2d υλοποιήσεων των σχημάτων και των διατάξεων, καθώς επίσης και με τη χρήση ειδικών νέων σχημάτων ή διατάξεων όποτε χρειάζεται.

Ζωγραφική

Συνοπτικά, η διαδικασία με την οποία ένα σχήμα ζωγραφίζεται είναι η εξής:

Figure#paint() : Αυτή η μέθοδος ξεκινά τη διαδικασία σχεδίασης, καθορίζοντας κάποιες ιδιότητες όπως η γραμματοσειρά και το χρώμα σχεδίασης του υπόβαθρου και του πρώτου πλάνου, και στη συνέχεια καλούνται οι εξής μέθοδοι:

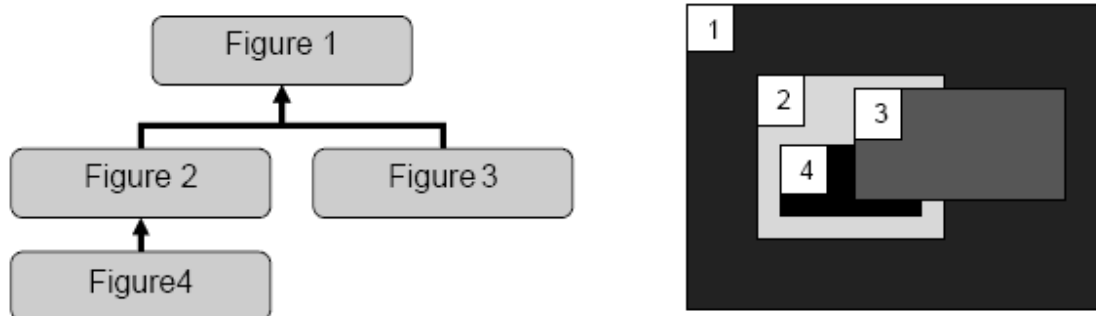
Figure#paintFigure(): Το σχήμα ζωγραφίζει τον εαυτό του. Μια απλή μέθοδος ζωγραφικής θα ήταν να γεμίζει την περιοχή μέσα στα όριά του με το χρώμα του υποβάθρου του.

Figure#paintClientArea(): Η περιοχή πελάτη είναι αυτή όπου πρόκειται (και επιτρέπεται) να εμφανίζονται τα παιδιά. Είναι πιθανό να γίνονται σε αυτήν τη μέθοδο μεταβολές στα γραφικά οι οποίες επηρεάζουν μόνο τα παιδιά, όπως αλλαγές που αφορούν το σύστημα συντεταγμένων.

Figure#paintChildren(): Μετά την ετοιμασία της περιοχής πελάτη, ζωγραφίζονται τα παιδιά.

Figure#paintBorder(): Τέλος, το σχήμα ζωγραφίζει διακοσμήσεις που πιθανόν να χρειάζεται να εμφανίζονται πάνω από τα παιδιά, και, αν έχει ρυθμιστεί έτσι, ζωγραφίζει και τα σύνορά του.

Το αποτέλεσμα αυτής της διαδικασίας είναι μια σύνθεση σχημάτων η οποία είναι δομημένη ως ένα δέντρο και ζωγραφίζεται με σειρά preorder και depth-first, με εξαίρεση τα σύνορα, που ζωγραφίζονται με σειρά postorder (Εικ. 74).



Εικ. 74. Ένα δέντρο σχημάτων και η γραφική αναπαράστασή του αν κάθε σχήμα ζωγραφιστεί ως πλήρες ορθογώνιο παραλληλόγραμμο

Συστήματα συντεταγμένων

Το Draw2d χρησιμοποιεί δύο διαφορετικά συστήματα συντεταγμένων. Το προκαθορισμένο σύστημα συντεταγμένων ονομάζεται απόλυτο και είναι πολύ απλό αφού είναι το ίδιο για όλα τα σχήματα. Παράλληλα, υπάρχει και η δυνατότητα να χρησιμοποιείται σχετικό (ή τοπικό) σύστημα συντεταγμένων, στο οποίο τα όρια κάθε παιδιού θεωρούνται σχετικά με την περιοχή πελάτη (δηλαδή τα όρια) του γονέα τους.

Για περιστάσεις όπου χρειάζεται να βρεθεί ένα σχήμα για μια θέση του δείκτη του ποντικιού, το Draw2d παρέχει τις κατάλληλες μεθόδους όπως την `findFigureAt(Point p)`.

Συνδέσεις

Τέλος, το Draw2d χρησιμοποιεί ειδικά σχήματα για να εμφανίζει γραμμές (συνδέσεις) μεταξύ δύο σημείων. Τα σχήματα αυτά υλοποιούν τη διαπροσωπεία `Connection`. Συνήθως οι συνδέσεις πρέπει να εμφανίζονται σε ένα στρώμα πάνω από τα άλλα σχήματα. Το αρχικό και το τελικό σημείο της γραμμής ορίζονται με χρήση σημείων άγκυρας (`ConnectionAnchor`), και τα υπόλοιπα σημεία της γραμμής σχεδιάζονται από έναν δρομολογητή (`ConnectionRouter`). Σε αντίθεση με τα άλλα σχήματα, τα όρια των συνδέσεων δεν καθορίζονται από τους γονείς τους ή από άλλες εξωτερικές οντότητες, αλλά υπολογίζονται από τον δρομολογητή. Ο απλούστερος δρομολογητής ενώνει το αρχικό και το τελικό σημείο με ένα ευθύγραμμο τμήμα, ενώ ένας άλλος δρομολογητής μπορεί να επιλέγει διαφορετικό τρόπο, λόγω χάρη να ορίζει τεθλασμένες γραμμές, ή οτιδήποτε άλλο.



Ως σχήματα, οι συνδέσεις μπορούν να έχουν και αυτές παιδιά. Τα παιδιά αυτά συνήθως είναι διακοσμήσεις, όπως για παράδειγμα ένα βέλος που δείχνει την κατεύθυνση της σύνδεσης ή μια ετικέτα κειμένου που την περιγράφει.

GEF

Γενικά

Το Draw2d εστιάζει στην αποδοτική διάταξη και σχεδίαση σχημάτων. Το plug-in GEF προσθέτει δυνατότητες επεξεργασίας πάνω στο Draw2d και ο σκοπός του είναι να:

1. Διευκολύνει την εμφάνιση οποιουδήποτε μοντέλου με γραφικό τρόπο χρησιμοποιώντας σχήματα του Draw2d.
2. Υποστηρίζει διαδραστικότητα μέσα από το ποντίκι, το πληκτρολόγιο ή τον πάγκο εργασίας.
3. Παρέχει έτοιμες κάποιες συχνά χρησιμοποιούμενες ψηφίδες που σχετίζονται με τα παραπάνω.

Η εικ. 75. δείχνει μια όψη του GEF, το οποίο μπορεί να οριστεί περίπου ως η περιοχή που βρίσκεται στη μέση. Είναι αυτό που παρέχει τη σύνδεση ανάμεσα στο μοντέλο και στην όψη μιας εφαρμογής.

Επίσης παρέχει χειριστές, όπως τα εργαλεία (tools) και οι ενέργειες (actions), οι οποίοι μετατρέπουν συμβάντα (events) σε αιτήματα (requests). Τα αιτήματα και οι εντολές (commands) χρησιμοποιούνται για την ενθυλάκωση των διαδράσεων και των αποτελεσμάτων τους πάνω στο μοντέλο, αντίστοιχα.

Μια εφαρμογή βασισμένη στο GEF χρησιμοποιεί αρχιτεκτονική MVC. Ο τρόπος με τον οποίο οι ρόλοι της αρχιτεκτονικής MVC εφαρμόζονται εδώ είναι ο εξής: Ο ορισμός του μοντέλου δεν αλλάζει, είναι τα δεδομένα που αποθηκεύονται, και αυτό προκύπτει από το πεδίο της εφαρμογής. Οι εντολές είναι ο τρόπος με τον οποίο το μοντέλο μεταβάλλεται, κατά έναν τρόπο που να μπορεί να γίνεται αναίρεση και επανάληψη των αλλαγών από το χρήστη. Η όψη είναι οτιδήποτε ορατό στο χρήστη, και τα στοιχεία της είναι είτε σχήματα του Draw2d (Figures), είτε στοιχεία ενός δέντρου (TreeItems). Μπορεί μια εφαρμογή να χρησιμοποιεί όψεις και των δύο ειδών ταυτόχρονα. Ο ελεγκτής στο GEF ονομάζεται EditPart και δεν είναι ένας, αλλά υπάρχει συνήθως ένας ελεγκτής για



κάθε στοιχείο του μοντέλου που έχει οπτική αναπαράσταση. Οι ελεγκτές είναι ο σύνδεσμος ανάμεσα στο μοντέλο και την όψη, και είναι υπεύθυνοι και για την επεξεργασία του μοντέλου. Τα EditParts περιέχουν βοηθητικά στοιχεία που ονομάζονται EditPolicies, τα οποία χειρίζονται το μεγαλύτερο μέρος της επεξεργασίας. Τα EditParts εμφανίζουν τις όψεις τους μέσα σε στοιχεία EditPartViewer. Υπάρχουν δύο τύποι τέτοιων στοιχείων, οι γραφικοί viewers, που εμφανίζουν Figures, και οι δενδρικοί viewers, που εμφανίζουν TreeItems.

EditParts

Τα EditParts συσχετίζουν το μοντέλο με την όψη τους, αλλά σχηματίζουν και τα ίδια μια δενδρική δομή. Κάθε editpart έχει παιδιά και συνήθως υπάρχει μια αντιστοιχία σχέση στο μοντέλο. Η σχέση γονέα – παιδιού των editparts επεκτείνεται και στα σχήματά τους, όπου το σχήμα του γονέα περιέχει τα σχήματα των παιδιών. Το αποτέλεσμα είναι τρεις διαφορετικές ιεραρχικές δομές δεδομένων, οι οποίες είναι περίπου παράλληλες η μία με την άλλη (εικ. 76).

Η εξαίρεση σε αυτήν την απλή δενδρική δομή είναι οι συνδέσεις (Connections), οι οποίες αντιπροσωπεύουν ένα σύνδεσμο ή μια συσχέτιση μεταξύ δύο αντικειμένων. Υπάρχουν δύο διαφορετικές υλοποιήσεις των editparts που παρέχει το GEF. Τα γραφικά editparts, που για όψη τους έχουν ένα σχήμα (figure) του Draw2d, υποστηρίζουν και τη δημιουργία συνδέσεων, οι οποίες είναι και αυτές γραφικά editparts. Τα δενδρικά editparts έχουν για όψη τους ένα αντικείμενο SWT TreeItem.

Οι ευθύνες ενός editpart είναι οι εξής:

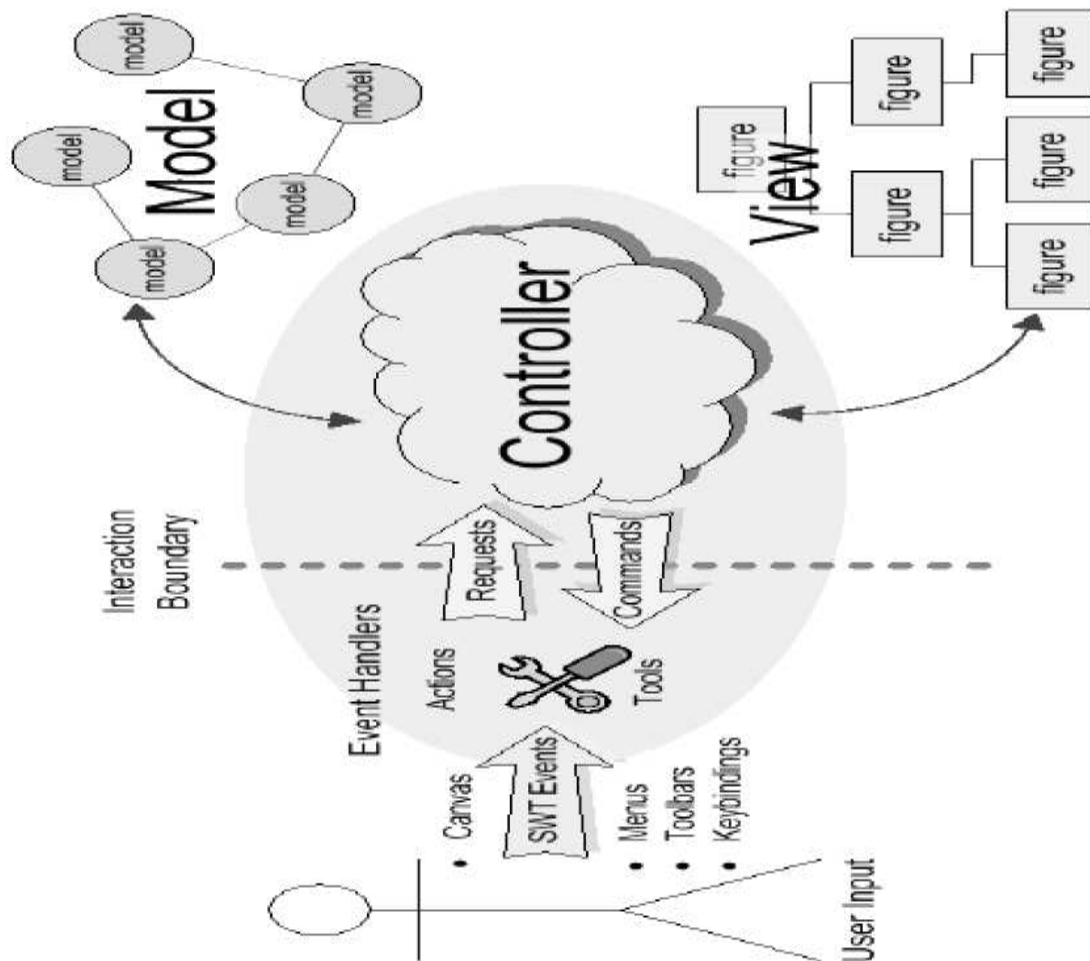
- Η δημιουργία και η διατήρηση μιας όψης (είτε figure είτε treeitem)
- Η δημιουργία και η διατήρηση των editparts – παιδιών του
- Η δημιουργία και η διατήρηση των editparts – συνδέσεων
- Η υποστήριξη της επεξεργασίας του μοντέλου

Η διατήρηση της όψης και των editparts προϋποθέτει ότι το editpart θα ενημερώνεται όταν γίνονται αλλαγές στο μοντέλο. Συνήθως αυτό γίνεται με την προσθήκη από το editpart ενός κατάλληλου listener (“ακροατή”) πάνω στο αντικείμενο του μοντέλου με το οποίο συσχετίζεται. Όταν λάβει μια ειδοποίηση ότι έγινε κάποια αλλαγή, ανανεώνει την όψη ή τη δομή του, ανάλογα με το είδος της αλλαγής.

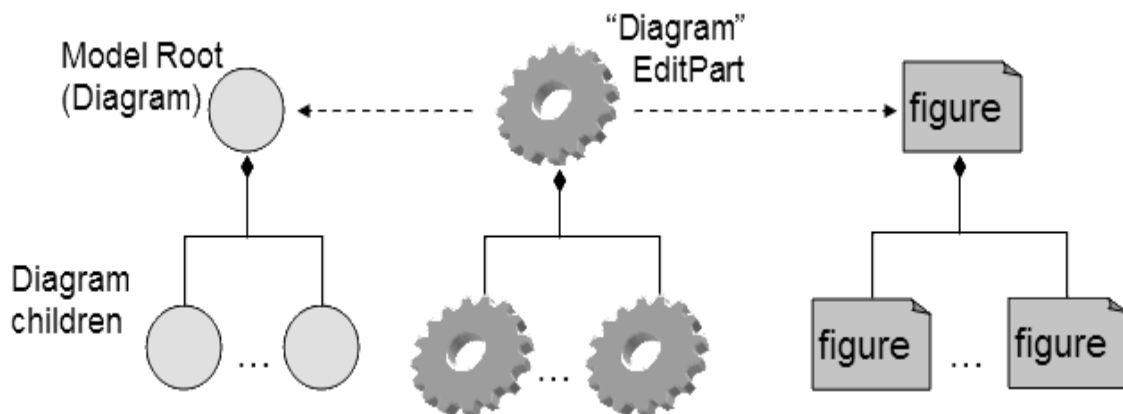


Δημιουργία γραφικής όψης του μοντέλου

Αν έχουμε ένα μοντέλο και μερικά σχήματα (figures), το επόμενο βήμα είναι να τα εμφανίσουμε στο χρήστη. Οι περισσότερες εφαρμογές χρησιμοποιούν ένα αντικείμενο της κλάσης ScrollingGraphicalViewer, που παρέχει το GEF για αυτό το σκοπό. Η κλάση αυτή είναι μια υλοποίηση ενός viewer που χρησιμοποιεί έναν καμβά FigureCanvas του Draw2d. Το επόμενο βήμα είναι να ορίσουμε το ριζικό editpart που θα χρησιμοποιήσουμε, το οποίο συνήθως είναι ένα ScalableRootEditPart και παρέχει κάποιες τυπικές βοηθητικές λειτουργίες, για παράδειγμα δυνατότητα zoom in / zoom out.



Εικ. 75. Η αρχιτεκτονική MVC όπως υλοποιείται από το GEF



Εικ. 76. Δενδρική δομή των EditParts

Στη συνέχεια ορίζονται τα editparts – παιδιά. Τα editparts αυτά είναι υποκλάσεις της βασικής κλάσης AbstractGraphicalEditPart από την οποία κληρονομούν τις απαραίτητες ιδιότητες και συμπεριφορά, και την οποία επεκτείνουν κατάλληλα, υπερσκελίζοντας απαραίτητα τις ακόλουθες μεθόδους:

createFigure() : μέθοδος που δημιουργεί την όψη (ή αλλιώς σχήμα) ενός editpart.

refreshVisuals(): Η μέθοδος αυτή αντανακλά τις ιδιότητες του μοντέλου πάνω στην όψη. Η μέθοδος αυτή καλείται κατά τη δημιουργία του editpart, αλλά και όταν γίνεται κάποια αλλαγή στο μοντέλο, αν χρειάζεται να ανανεωθεί η όψη για να είναι συνεπής με τη νέα κατάσταση του μοντέλου.

getModelChildren(): Η μέθοδος αυτή επιστρέφει μια λίστα με τα παιδιά του αντίστοιχου αντικειμένου στο μοντέλο. Χρειάζεται ώστε να μπορεί το editpart να δημιουργεί και τα παιδιά του.

getModelSourceConnections(), getModelTargetConnections(): Οι δύο αυτές μέθοδοι πρέπει να υπερσκελίζονται από ένα editpart, αν αυτό έχει τη δυνατότητα να είναι αρχή ή τέλος μιας σύνδεσης.

Επιστρέφουν τα αντικείμενα του μοντέλου που αντιπροσωπεύουν τις συνδέσεις.

Τα editparts – συνδέσεις είναι ειδικά editparts που συνδέουν οποιαδήποτε δύο editparts μέσα σε ένα διάγραμμα. Ένα editpart ονομάζεται κόμβος (NodeEditPart) αν μπορεί να είναι αρχή ή τέλος μιας σύνδεσης. Τα σχήματα των editparts – συνδέσεων είναι ειδικά σχήματα (συνδέσεις του Draw 2d) αλλά κατά τα άλλα οι συνδέσεις μοιάζουν με τα άλλα



editparts στο ότι μπορεί να έχουν και αυτές παιδιά και ιδιότητες που πρέπει να αντανακλώνται στην όψη τους.

Επεξεργασία

Η επεξεργασία είναι συνήθως η πιο περίπλοκη εργασία που κάνει ένα editpart και συμπεριλαμβάνει την πραγματοποίηση αλλαγών στο μοντέλο και την εμφάνιση γραφικής ανατροφοδότησης την ώρα που ο χρήστης αλληλεπιδρά με την όψη.

Η πηγή μιας αλληλεπίδρασης αναπαριστάνεται με ένα αίτημα (request), το οποίο δημιουργείται από ένα εργαλείο όπως εξηγείται παρακάτω. Το πρώτο βήμα της επεξεργασίας είναι να αποφασίσει το σύστημα ποια editparts επηρεάζονται. Αυτά συνήθως είναι τα επιλεγμένα από το χρήστη editparts ή το editpart που βρίσκεται στην τρέχουσα θέση του δείκτη ποντικιού, το οποίο μπορεί να βρεθεί με τη μέθοδο `getTargetEditPart (Request)` του `EditPartViewer`. Η εντολή (Command) είναι αυτή που τελικά μεταβάλλει το μοντέλο. Τα editparts είναι κατασκευασμένα ώστε να επιστρέφουν μια εντολή για ένα αίτημα που τους δίνεται. Αν ένα editpart δεν επιστρέφει εντολή για ένα αίτημα, αυτό σημαίνει ότι δεν επιτρέπεται αυτό που ζητήθηκε (για παράδειγμα, μπορεί να απαγορεύεται η διαγραφή κάποιου συγκεκριμένου στοιχείου) και η γραφική διαπροσωπεία δείχνει μια κατάλληλη ένδειξη στο χρήστη.

Τα editparts δε χειρίζονται απευθείας την επεξεργασία, αλλά την αναθέτουν σε πολιτικές επεξεργασίας (EditPolicies). Κάθε EditPolicy εστιάζει σε μια εργασία ή σε μια ομάδα συναφών εργασιών και υλοποιεί μεθόδους `getCommand(Request)` για να επιστρέφει την κατάλληλη εντολή για ένα δοθέν αίτημα. Κάθε EditPart εγκαθιστά τις EditPolicies που χρειάζεται κατά την ώρα της δημιουργίας του. Οι εντολές χρησιμοποιούνται αντί της απευθείας επεξεργασίας του μοντέλου με σκοπό να ενθυλακώνουν και να συνδυάζουν τις αλλαγές πάνω στο μοντέλο της εφαρμογής. Κάθε εφαρμογή έχει μια μοναδική στοίβα εντολών, και κάθε φορά που εκτελείται μια εντολή, αυτή προστίθεται στη στοίβα.

Κάθε εντολή υλοποιεί τις μεθόδους `undo()` και `redo()` ώστε ο χρήστης να μπορεί να κάνει αναίρεση και επανάληψη των τελευταίων αλλαγών που έκανε.



Εργαλεία και παλέτα εργαλείων

Τα εργαλεία αναλαμβάνουν το χειρισμό των συμβάντων. Μπορεί μια εφαρμογή να έχει πολλαπλά εργαλεία, τα οποία εμφανίζονται συνήθως σε μια παλέτα δίπλα στον καμβά, αλλά μόνο ένα από αυτά είναι ενεργό κάθε στιγμή. Η παλέτα χρησιμοποιείται για να επιλέγει ο χρήστης το ενεργό εργαλείο.

Τα εργαλεία επιτελούν λειτουργίες όπως:

- Ζητούν από τα editparts να δείξουν ανατροφοδότηση σχετικά με το τρέχον αίτημα
- Ζητούν και παίρνουν εντολές από τα editparts
- Εκτελούν μια εντολή στη στοίβα εντολών
- Ανανεώνουν κατάλληλα το δείκτη του ποντικιού

Το προεπιλεγμένο εργαλείο είναι συνήθως το εργαλείο επιλογής (Selection Tool).

Τύποι διαδράσεων

Ο χρήστης μιας εφαρμογής βασισμένης στο GEF μπορεί να αλληλεπιδρά με αυτήν με διάφορους τρόπους. Με τον όρο αλληλεπίδραση εννοούμε οτιδήποτε επηρεάζει το μοντέλο ή την κατάσταση της γραφικής διαπροσωπείας. Πολλές αλληλεπιδράσεις είναι γραφικές, αλλά μερικές δεν είναι, παραδείγματα:

- Πρόκληση κάποιας ενέργειας (Action) (συνήθως από μια γραμμή μενού, μια γραμμή εργαλείων, ή ένα μενού δεξιού κλικ)
- Πάτημα του ποντικιού πάνω σε κάτι
- Σύρσιμο κάποιου αντικειμένου με το ποντίκι
- Άφημα κάποιου αντικειμένου που σύρθηκε (λειτουργία Drag-N-Drop)
- Πάτημα κάποιων πλήκτρων

Υπάρχουν κάποιες βασικές λειτουργίες που μπορεί να εκτελεί ο χρήστης πάνω στο μοντέλο, οι οποίες είναι κοινές για πολλές εφαρμογές και γι αυτό το GEF παρέχει έτοιμες ευκολίες για την υλοποίησή τους. Για παράδειγμα, αν έχουμε έναν γραφικό editor που επεξεργάζεται ψηφιακά λογικά κυκλώματα, μερικές βασικές λειτουργίες θα ήταν η δημιουργία μιας νέας λογικής πύλης, η δημιουργία μιας σύνδεσης μεταξύ δύο πυλών, η μετακίνηση μιας πύλης στο χώρο, η αλλαγή των διαστάσεων ενός οπτικού αντικειμένου, η διαγραφή μιας πύλης κ.ο.κ.



Εξετάζοντας λίγο πιο αναλυτικά το παράδειγμα της διαγραφής ενός στοιχείου, έστω ότι ο χρήστης έχει επιλέξει, με χρήση του εργαλείου επιλογής, μια λογική πύλη (ή μπορεί και περισσότερες). Για να τη διαγράψει από το διάγραμμα θα πρέπει να κάνει μια ενέργεια διαγραφής (DeleteAction), κάτι που μπορεί να γίνει είτε από τη γραμμή μενού, είτε από το μενού δεξιού κλικ, είτε από το πληκτρολόγιο με το πλήκτρο Del ή και κάπως αλλιώς. Όπως και αν γίνει η ενέργεια, το αποτέλεσμα είναι να σταλεί ένα αίτημα διαγραφής με προκαθορισμένο όνομα REQ_DELETE στο editpart της επιλεγμένης πύλης. Όλα τα editparts πρέπει να έχουν μια editpolicy που είτε υποστηρίζει είτε απαγορεύει τη διαγραφή του αντίστοιχου αντικειμένου. Μια ορθή και πλήρης υλοποίηση της εντολής διαγραφής θα πρέπει να λαμβάνει υπόψην της αν το αντικείμενο που πρόκειται να διαγραφεί έχει συνδέσεις ή αν τα παιδιά του έχουν συνδέσεις, και να φροντίζει να διαγράφονται και αυτές τις συνδέσεις.



Παράρτημα Γ : Γλώσσα XML

Η γλώσσα XML (eXtensible Markup Language) [41] είναι μια προδιαγραφή γενικής χρήσης για τη δημιουργία γλωσσών σήμανσης (markup). Χαρακτηρίζεται επεκτάσιμη επειδή επιτρέπει στο χρήστη να ορίζει τα στοιχεία της σήμανσης. Ο σκοπός της XML είναι να βοηθά τα συστήματα πληροφοριών να αποθηκεύουν και να διαμοιράζονται δομημένα δεδομένα.

Η XML είναι μια πρόταση του World Wide Web Consortium. Είναι μια ανοιχτή προδιαγραφή, η οποία ορίζει κανόνες λεκτικής και συντακτικής ανάλυσης των εγγράφων. Μια και η XML είναι η γλώσσα υλοποίησης των προδιαγραφών IMS Content Packaging και IMS Simple Sequencing, αλλά και του μηχανισμού επέκτασης της πλατφόρμας eclipse, παρουσιάζεται εδώ μια σύντομη περιγραφή της γλώσσας.

Ένα έγγραφο XML έχει δύο επίπεδα ορθότητας:

Ορθή μορφή : Ένα ορθά διαμορφωμένο έγγραφο υπακούει στους συντακτικούς κανόνες της XML (συντακτική ορθότητα). Για παράδειγμα, αν υπάρχει κάπου μια ετικέτα εκκίνησης (< >) χωρίς αντίστοιχη ετικέτα τερματισμού (</>), το έγγραφο δεν έχει τη σωστή μορφή και δεν είναι έγγραφο XML.

Εγκυρότητα : Ένα έγκυρο έγγραφο υπακούει επιπλέον σε κάποιους σημασιολογικούς κανόνες (σημασιολογική ορθότητα), είτε ορισμένους από το χρήστη, είτε μέσα σε ένα σχήμα XML, ή σε έναν ορισμό τύπου εγγράφου. Για παράδειγμα, αν ένα έγγραφο περιέχει ένα μη ορισμένο στοιχείο, τότε το έγγραφο δεν είναι έγκυρο.

Ορθή μορφή

Είναι δυνατό σε κάποιες περιπτώσεις να μη γίνεται κανένας έλεγχος εγκυρότητας, οπότε η γλώσσα XML χρησιμοποιείται ως ένα γενικό πλαίσιο για την αποθήκευση οποιασδήποτε ποσότητας κειμένου ή άλλων δεδομένων, των οποίων η δομή μπορεί να αναπαρασταθεί με ένα δέντρο.

Η βασική σύνταξη ενός στοιχείου (element) XML είναι η εξής:

<όνομαΣτοιχείου όνομαΙδιότητας="τιμή ιδιότητας">

Περιεχόμενο στοιχείου

</όνομαΣτοιχείου>



Τα δύο στιγμιότυπα του ονόματος του στοιχείου αναφέρονται σαν ετικέτα εκκίνησης (start-tag) και ετικέτα τερματισμού (end-tag) αντίστοιχα. Το περιεχόμενο του στοιχείου είναι κάποιο κείμενο, που μπορεί να συμπεριλαμβάνει και άλλα στοιχεία XML, σχηματίζοντας έτσι μια δενδρική δομή δεδομένων. Κάθε στοιχείο μπορεί να έχει πολλές ιδιότητες. Μπορούν να τοποθετηθούν οπουδήποτε μέσα στη δενδρική δομή σχόλια, που ξεκινούν με `<!--` και τελειώνουν σε `-->`. Οι όροι γονέας, παιδί και αδέρφια χρησιμοποιούνται για να περιγράψουν τις σχέσεις μεταξύ των στοιχείων.

Στο παράδειγμα εγγράφου XML του κώδικα Α.1, η πρώτη γραμμή είναι η δήλωση XML, που ορίζει την έκδοση της γλώσσας (1.0) και την κωδικοποίηση χαρακτήρων που χρησιμοποιείται (ISO-8859-1).

Η επόμενη γραμμή (`<note>`) περιγράφει το ριζικό στοιχείο (root element) του εγγράφου, και ουσιαστικά λέει ότι αυτό το έγγραφο είναι ένα σημείωμα. Το ριζικό στοιχείο έχει και μια ιδιότητα `time` που το περιγράφει, με τιμή 9.30. Οι επόμενες τέσσερις γραμμές περιέχουν τέσσερα στοιχεία παιδιά του ριζικού στοιχείου (`to`, `from`, `heading`, `body`). Η τελευταία γραμμή (`</note>`) σηματοδοτεί το τέλος του ριζικού στοιχείου. Θα μπορούσε κάποιο παιδί, εκτός από κείμενο, να έχει και δικά του παιδιά.

Πηγαίος Κώδικας Α.1: Παράδειγμα εγγράφου XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note time="9.30">
<to>Chris</to>
<from>Alex</from>
<heading>Greeting</heading>
<body>Good morning</body>
</note>
```

Μπορούμε να υποθέσουμε, χωρίς να έχουμε άλλα στοιχεία, ότι το παραπάνω έγγραφο περιγράφει ένα σημείωμα από τον Alex στον Chris. Γενικά, τα έγγραφα της XML έχουν τη χρήσιμη ιδιότητα να περιέχουν αρκετές πληροφορίες ώστε να περιγράφουν τον εαυτό τους.



Εγκυρότητα

Ένα έγκυρο έγγραφο XML είναι ένα “ορθά διαμορφωμένο” έγγραφο XML, σύμφωνα με αυτά που περιγράφηκαν νωρίτερα, το οποίο επιπλέον υπακούει στους κανόνες ενός ορισμού Document Type Definition (DTD). Στο παράδειγμα Α.2 φαίνεται πώς, με χρήση της δήλωσης DOCTYPE, έχουμε αναφορά σε ένα εξωτερικό αρχείο DTD, το περιεχόμενο του οποίου φαίνεται στον κώδικα Α.3.

Πηγαίος Κώδικας Α.2: Έγγραφο XML με αναφορά σε αρχείο DTD

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE note SYSTEM "Note.dtd">
<note time="9.30">
<!-- ... -->
```

Ο σκοπός του αρχείου DTD είναι να ορίσει τη δομή ενός εγγράφου XML, και αυτό γίνεται με μια λίστα από επιτρεπτά στοιχεία.

Πηγαίος Κώδικας Α.3: Αρχείο DTD

```
<!DOCTYPE note
[
<!ELEMENT note ( to , from , heading , body ) >
<!ELEMENT to ( #PCDATA ) >
<!ELEMENT from ( #PCDATA ) >
<!ELEMENT heading ( #PCDATA ) >
<!ELEMENT body ( #PCDATA ) >
]>
```

Όπου PCDATA (parsed character data) είναι κείμενο το οποίο υφίσταται συντακτική ανάλυση από τον XML parser (υπάρχει και η δυνατότητα κάποια τμήματα κειμένου, τα οποία ονομάζονται CDATA, να εξαιρούνται από τη συντακτική ανάλυση).

Υπάρχει επίσης ένας εναλλακτικός τρόπος περιγραφής της δομής ενός εγγράφου XML, που ονομάζεται XML Schema, και βασίζεται στην ίδια τη γλώσσα XML (παράδειγμα Α.4). Η μέθοδος αυτή, που είναι γνωστή και ως XSD (XML Schema Definition), θεωρείται διάδοχος του DTD επειδή είναι πιο ισχυρή στην περιγραφή γλωσσών XML.



Πηγαίος Κώδικας A.4: Αρχείο XSD

```
< xs : e l e m e n t name="note">  
< xs : complexType >  
  < xs : s e q u e n c e >  
    < xs : e l e m e n t name="to" t y p e ="xs:string"/ >  
    < xs : e l e m e n t name="from" t y p e ="xs:string"/ >  
    < xs : e l e m e n t name="heading" t y p e ="xs:string"/ >  
    < xs : e l e m e n t name="body" t y p e ="xs:string"/ >  
  </ xs : s e q u e n c e >  
</ xs : complexType >  
</ xs : e l e m e n t >
```



Παράρτημα Δ: Τα.ecore μεταμοντέλα της AMOLA (αρχεία .xmi)

Λίστα 1. Το SAG μεταμοντέλο σε XML φορμά (αρχείο SAG.ecore)

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="SAG"
  nsURI="http://www.amcl.tuc.gr/aseme/metamodels/SAG"
  nsPrefix="SAG">
  <eClassifiers xsi:type="ecore:EClass" name="SAGmodel">
    <eStructuralFeatures xsi:type="ecore:EReference" name="goals"
      lowerBound="1" upperBound="-1"
      eType="#//Goal" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="actors"
      lowerBound="1"
      upperBound="-1" eType="#//Actor" containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Actor">
    <eStructuralFeatures xsi:type="ecore:EReference" name="my_goal"
      upperBound="-1"
      eType="#//Goal" eOpposite="#//Goal/dependee"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"
      defaultValueLiteral=""/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Goal">
    <eStructuralFeatures xsi:type="ecore:EAttribute"
      name="requirements" eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="dependee" lowerBound="1"
      eType="#//Actor"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="dependee" upperBound="-1"
      eType="#//Actor" eOpposite="#//Actor/my_goal"/>
  </eClassifiers>
</ecore:EPackage>
```



Λίστα 2. Το SUC μεταμοντέλο σε XML φορμά (αρχείο SUC.ecore)

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="SUC"
  nsURI="http://www.amcl.tuc.gr/aseme/metamodels/SUC"
  nsPrefix="SUC">
  <eClassifiers xsi:type="ecore:EClass" name="SUCmodel">
    <eStructuralFeatures xsi:type="ecore:EReference" name="roles"
      lowerBound="2" upperBound="-1"
      eType="#//Role" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="usecases" lowerBound="1"
      upperBound="-1" eType="#//UseCase" containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="UseCase">
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="participant" ordered="false"
      upperBound="-1" eType="#//Role"
      eOpposite="#//Role/participates_in"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="include"
      ordered="false"
      upperBound="-1" eType="#//UseCase"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
      name="specified_by" eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Role">
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="participates_in" ordered="false"
      upperBound="-1" eType="#//UseCase"
      eOpposite="#//UseCase/participant"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
      http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="HumanRole"
    eSuperTypes="#//Role"/>
  <eClassifiers xsi:type="ecore:EClass" name="SystemRole"
    eSuperTypes="#//Role"/>
</ecore:EPackage>
```




Λίστα 3. Το AIP μεταμοντέλο σε XML φορμά (αρχείο AIP.ecore)

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="AIP"
  nsURI="http://www.acml.tuc.gr/aseme/metamodels/AIP"
  nsPrefix="AIP">
  <eClassifiers xsi:type="ecore:EClass" name="AIPmodel">
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="participants" lowerBound="2"
        upperBound="-1" eType="#//Participant"
      containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="protocols" lowerBound="1"
        upperBound="-1" eType="#//Protocol" containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Protocol">
    <eStructuralFeatures xsi:type="ecore:EReference"
      name="participants" lowerBound="2"
        upperBound="-1" eType="#//Participant"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Participant">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
      eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"
        defaultValueLiteral=""/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
      name="engaging_rules" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"
        defaultValueLiteral=""/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
      name="outcomes" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"
        defaultValueLiteral=""/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
      name="liveness" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
</ecore:EPackage>
```



Λίστα 4. Το SRM μεταμοντέλο σε XML φορμά (αρχείο SRM.ecore)

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="SRM"
  nsURI="http://www.amcl.tuc.gr/aseme/metamodels/SRM"
  nsPrefix="SRM">
  <eClassifiers xsi:type="ecore:EClass" name="SRMmodel">
    <eStructuralFeatures xsi:type="ecore:EReference"
name="activities" upperBound="-1"
    eType="#//Activity" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
name="capabilities" upperBound="-1"
    eType="#//Capability" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="roles"
lowerBound="1" upperBound="-1"
    eType="#//Role" containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Role">
    <eStructuralFeatures xsi:type="ecore:EReference"
name="activities" ordered="false"
    lowerBound="1" upperBound="-1" eType="#//Activity"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="liveness"
ordered="false"
    lowerBound="1" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
lowerBound="1" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
name="capabilities" ordered="false"
    upperBound="-1" eType="#//Capability"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Activity">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
ordered="false" unique="false"
    lowerBound="1" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
name="functionality" ordered="false"
    unique="false" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Capability">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
ordered="false" unique="false"
    lowerBound="1" eType="ecore:EDatatype
```



```
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  <eStructuralFeatures xsi:type="ecore:EReference"
name="activities" upperBound="-1"
  eType="#//Activity"/>
</eClassifiers>
</ecore:EPackage>
```

Λίστα 5. Το IAC μεταμοντέλο σε XML φορμά (αρχείο IAC.ecore)

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="IAC"
  nsURI="http://www.amcl.tuc.gr/aseme/metamodels/IAC"
  nsPrefix="IAC">
  <eClassifiers xsi:type="ecore:EClass" name="Node">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="type"
eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="label"
eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute"
name="activity" eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
name="variables" upperBound="-1"
  eType="#//Variable"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
name="Father_of" upperBound="-1"
  eType="#//Node"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Variable">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="type"
eType="ecore:EDatatype
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Transition">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="TE"
eType="ecore:EDatatype
```



```
    http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="source"
lowerBound="1"
    eType="#//Node"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="target"
lowerBound="1"
    eType="#//Node"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
eType="ecore:EDataType
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    </eClassifiers>
    <eClassifiers xsi:type="ecore:EClass" name="Model">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name"
eType="ecore:EDataType
http://www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference"
name="variables" upperBound="-1"
    eType="#//Variable" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="nodes"
upperBound="-1"
    eType="#//Node" containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="transitions"
upperBound="-1"
    eType="#//Transition" containment="true"/>
    </eClassifiers>
</ecore:EPackage>
```



Παράρτημα Ε: 'Το σύστημα διαχείρισης συναντήσεων' (αρχεία .xmi)

Λίστα 6. Το SAG μοντέλο σε XML φορμά (αρχείο SAG_example.xmi)

```
<?xml version="1.0" encoding="UTF-8"?>
<notation:Diagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SAG="http://www.amcl.tuc.gr/aseme/metamodels/SAG"
xmlns:notation="http://www.eclipse.org/gmf/runtime/1.0.2/notation"
xmi:id="_iHfMka70Ed-Fi8rA7Di7ZA" type="SAG" name="SAG_examle.sagd"
measurementUnit="Pixel">
  <children xmi:type="notation:Node" xmi:id="_q1_NwK70Ed-Fi8rA7Di7ZA"
type="2007">
    <children xmi:type="notation:DecorationNode" xmi:id="_q1_NxK70Ed-
Fi8rA7Di7ZA" type="5010"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_q1_Nwa70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_q1_Nwq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Actor" href="SAG_examle.sag#//@actors.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_q1_Nw670Ed-
Fi8rA7Di7ZA" x="15" y="280" width="51" height="61"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_tMo8EK70Ed-Fi8rA7Di7ZA"
type="2007">
    <children xmi:type="notation:DecorationNode" xmi:id="_tMo8FK70Ed-
Fi8rA7Di7ZA" type="5010"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_tMo8Ea70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_tMo8Eq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Actor" href="SAG_examle.sag#//@actors.1"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_tMo8E670Ed-
Fi8rA7Di7ZA" x="145" y="165" width="118" height="81"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_208qEK70Ed-Fi8rA7Di7ZA"
type="2007">
    <children xmi:type="notation:DecorationNode" xmi:id="_208qFK70Ed-
Fi8rA7Di7ZA" type="5010"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_208qEa70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_208qEq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Actor" href="SAG_examle.sag#//@actors.2"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_208qE670Ed-
Fi8rA7Di7ZA" x="460" y="175" width="66" height="61"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_4kQ7oK70Ed-Fi8rA7Di7ZA"
type="2008">
    <children xmi:type="notation:DecorationNode" xmi:id="_4kQ7pK70Ed-
Fi8rA7Di7ZA" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_4kQ7pa70Ed-
Fi8rA7Di7ZA" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_4kQ7oa70Ed-
```



```
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_4kQ7oq70Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_4kQ7o670Ed-
Fi8rA7Di7ZA" x="5" y="170" width="91"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_5nBZIK70Ed-Fi8rA7Di7ZA"
type="2008">
    <children xmi:type="notation:DecorationNode" xmi:id="_5nBZJK70Ed-
Fi8rA7Di7ZA" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_5nKjEK70Ed-
Fi8rA7Di7ZA" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_5nBZia70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_5nBZIq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.1"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_5nBZI670Ed-
Fi8rA7Di7ZA" x="145" y="30" width="116"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_6Lh0YK70Ed-Fi8rA7Di7ZA"
type="2008">
    <children xmi:type="notation:DecorationNode" xmi:id="_6Lh0ZK70Ed-
Fi8rA7Di7ZA" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_6Lh0Za70Ed-
Fi8rA7Di7ZA" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_6Lh0Ya70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_6Lh0Yq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.2"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_6Lh0Y670Ed-
Fi8rA7Di7ZA" x="120" y="345" width="171"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_7IxfwK70Ed-Fi8rA7Di7ZA"
type="2008">
    <children xmi:type="notation:DecorationNode" xmi:id="_7IxfxK70Ed-
Fi8rA7Di7ZA" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_7Ixfxa70Ed-
Fi8rA7Di7ZA" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_7Ixfwa70Ed-
Fi8rA7Di7ZA"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_7Ixfwq70Ed-
Fi8rA7Di7ZA"/>
    <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.3"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_7Ixfw670Ed-
Fi8rA7Di7ZA" x="285" y="255" width="121"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_Affs0K71Ed-Fi8rA7Di7ZA"
type="2008">
    <children xmi:type="notation:DecorationNode" xmi:id="_Affs1K71Ed-
Fi8rA7Di7ZA" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_Affs1a71Ed-
```



```
Fi8rA7Di7ZA" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_Affs0a71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_Affs0q71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.4"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Affs0671Ed-
Fi8rA7Di7ZA" x="385" y="50" width="96"/>
</children>
<children xmi:type="notation:Node" xmi:id="_A7g6cK71Ed-Fi8rA7Di7ZA"
type="2008">
  <children xmi:type="notation:DecorationNode" xmi:id="_A7g6dK71Ed-
Fi8rA7Di7ZA" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_A7g6da71Ed-
Fi8rA7Di7ZA" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_A7g6ca71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_A7g6cq71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.5"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_A7g6c671Ed-
Fi8rA7Di7ZA" x="585" y="50" width="156"/>
</children>
<children xmi:type="notation:Node" xmi:id="_Dis_IK71Ed-Fi8rA7Di7ZA"
type="2007">
  <children xmi:type="notation:DecorationNode" xmi:id="_Dis_JK71Ed-
Fi8rA7Di7ZA" type="5010"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_Dis_Ia71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_Dis_Iq71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Actor" href="SAG_examle.sag#//@actors.3"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Dis_I671Ed-
Fi8rA7Di7ZA" x="649" y="375" width="76" height="66"/>
</children>
<children xmi:type="notation:Node" xmi:id="_JYRrYK71Ed-Fi8rA7Di7ZA"
type="2008">
  <children xmi:type="notation:DecorationNode" xmi:id="_JYRrZK71Ed-
Fi8rA7Di7ZA" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_JYRrZa71Ed-
Fi8rA7Di7ZA" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_JYRrYa71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_JYRrYq71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.6"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_JYRrY671Ed-
Fi8rA7Di7ZA" x="620" y="195" width="126"/>
</children>
<children xmi:type="notation:Node" xmi:id="_ML1YIK71Ed-Fi8rA7Di7ZA"
type="2008">
  <children xmi:type="notation:DecorationNode" xmi:id="_ML1YJK71Ed-
Fi8rA7Di7ZA" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_ML1YJa71Ed-
```




```
Fi8rA7Di7ZA" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_ML1YIa71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_ML1YIq71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.7"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_ML1YI671Ed-
Fi8rA7Di7ZA" x="430" y="360" width="121"/>
</children>
<children xmi:type="notation:Node" xmi:id="_Mxr3MK71Ed-Fi8rA7Di7ZA"
type="2008">
  <children xmi:type="notation:DecorationNode" xmi:id="_Mxr3NK71Ed-
Fi8rA7Di7ZA" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_Mxr3Na71Ed-
Fi8rA7Di7ZA" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_Mxr3Ma71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_Mxr3Mq71Ed-
Fi8rA7Di7ZA"/>
  <element xmi:type="SAG:Goal" href="SAG_examle.sag#//@goals.8"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Mxr3M671Ed-
Fi8rA7Di7ZA" x="530" y="285" width="121"/>
</children>
<styles xmi:type="notation:DiagramStyle" xmi:id="_iHfMkq70Ed-
Fi8rA7Di7ZA"/>
<element xmi:type="SAG:SAGmodel" href="SAG_examle.sag#/">
  <edges xmi:type="notation:Connector" xmi:id="_M4kF0K72Ed-Fi8rA7Di7ZA"
type="4011" source="_q1NwK70Ed-Fi8rA7Di7ZA" target="_4kQ7oK70Ed-
Fi8rA7Di7ZA">
    <styles xmi:type="notation:FontStyle" xmi:id="_M4kF0a72Ed-
Fi8rA7Di7ZA"/>
    <element xsi:nil="true"/>
    <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_M4kF0q72Ed-Fi8rA7Di7ZA" points="[2, -30, 0, 75]${-1, -100, -3,
5}"/>
    <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_M5UTwK72Ed-Fi8rA7Di7ZA" id="(0.4065934065934066,0.875)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_WHi94K72Ed-Fi8rA7Di7ZA"
type="4011" source="_tMo8EK70Ed-Fi8rA7Di7ZA" target="_7IXfwK70Ed-
Fi8rA7Di7ZA">
    <styles xmi:type="notation:FontStyle" xmi:id="_WHi94a72Ed-
Fi8rA7Di7ZA"/>
    <element xsi:nil="true"/>
    <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_WHi94q72Ed-Fi8rA7Di7ZA" points="[3, 1, -88, -57]${86, 55, -5, -
3}"/>
    <sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_Wnu50K72Ed-Fi8rA7Di7ZA"
id="(0.9745762711864406,0.43209876543209874)"/>
    <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_WHi94672Ed-Fi8rA7Di7ZA" id="(0.05785123966942149,0.975)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_hYnpgK72Ed-Fi8rA7Di7ZA"
```



```
type="4012" source="_4kQ7oK70Ed-Fi8rA7Di7ZA" target="_tMo8EK70Ed-  
Fi8rA7Di7ZA">  
  <styles xmi:type="notation:FontStyle" xmi:id="_hYnpgq72Ed-  
Fi8rA7Di7ZA"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_hYnpgq72Ed-Fi8rA7Di7ZA" points="[46, 2, -60, -4]${95, 3, -11, -  
3}"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_hYnpg672Ed-Fi8rA7Di7ZA"  
id="(0.09322033898305085,0.38271604938271603)"/>  
</edges>  
  <edges xmi:type="notation:Connector" xmi:id="_jXH9kK72Ed-Fi8rA7Di7ZA"  
type="4013" source="_5nBZIK70Ed-Fi8rA7Di7ZA" target="_tMo8EK70Ed-  
Fi8rA7Di7ZA">  
  <styles xmi:type="notation:FontStyle" xmi:id="_jXH9ka72Ed-  
Fi8rA7Di7ZA"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_jXH9kq72Ed-Fi8rA7Di7ZA" points="[-2, 20, 0, -96]${-3, 115, -1,  
-1}"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_jXH9k672Ed-Fi8rA7Di7ZA"  
id="(0.4745762711864407,0.012345679012345678)"/>  
</edges>  
  <edges xmi:type="notation:Connector" xmi:id="_kHaLAK72Ed-Fi8rA7Di7ZA"  
type="4013" source="_6Lh0YK70Ed-Fi8rA7Di7ZA" target="_tMo8EK70Ed-  
Fi8rA7Di7ZA">  
  <styles xmi:type="notation:FontStyle" xmi:id="_kHaLa72Ed-  
Fi8rA7Di7ZA"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_kHaLaq72Ed-Fi8rA7Di7ZA" points="[1, -9, -13, 104]${12, -108, -  
2, 5}"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_kHjU8K72Ed-Fi8rA7Di7ZA" id="(0.36257309941520466,0.225)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_kHjU8a72Ed-Fi8rA7Di7ZA"  
id="(0.4322033898305085,0.9382716049382716)"/>  
</edges>  
  <edges xmi:type="notation:Connector" xmi:id="_lRxFUK72Ed-Fi8rA7Di7ZA"  
type="4013" source="_Affs0K71Ed-Fi8rA7Di7ZA" target="_208qEK70Ed-  
Fi8rA7Di7ZA">  
  <styles xmi:type="notation:FontStyle" xmi:id="_lRxFUa72Ed-  
Fi8rA7Di7ZA"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_lRxFUq72Ed-Fi8rA7Di7ZA" points="[9, 20, -51, -115]${61, 105, 1,  
-30}"/>  
</edges>  
  <edges xmi:type="notation:Connector" xmi:id="_l100oK72Ed-Fi8rA7Di7ZA"  
type="4013" source="_A7g6cK71Ed-Fi8rA7Di7ZA" target="_208qEK70Ed-  
Fi8rA7Di7ZA">  
  <styles xmi:type="notation:FontStyle" xmi:id="_l100oa72Ed-
```



```
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_l100oq72Ed-Fi8rA7Di7ZA" points="[-26, 20, 122, -98]$[-137, 109,
11, -9]"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_l19-
kK72Ed-Fi8rA7Di7ZA" id="(0.9393939393939394,0.5737704918032787)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_oHjmsK72Ed-Fi8rA7Di7ZA"
type="4012" source="_7IxfwK70Ed-Fi8rA7Di7ZA" target="_208qEK70Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_oHjmsa72Ed-
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_oHjmsq72Ed-Fi8rA7Di7ZA" points="[10, -4, -121, 44]$[119, -44, -
12, 4]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_oHjms672Ed-Fi8rA7Di7ZA" id="(0.9752066115702479,0.575)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_oHjmtK72Ed-Fi8rA7Di7ZA"
id="(0.18181818181818182,0.5901639344262295)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_uZcbgK72Ed-Fi8rA7Di7ZA"
type="4011" source="_Dis_IK71Ed-Fi8rA7Di7ZA" target="_JYRrYK71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_uZcbga72Ed-
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_uZcbgq72Ed-Fi8rA7Di7ZA" points="[0, -33, -5, 144]$[7, -173, 2,
4]"/>
  <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_uZllcK72Ed-Fi8rA7Di7ZA" id="(0.5714285714285714,0.9)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_vhHOMK72Ed-Fi8rA7Di7ZA"
type="4011" source="_Dis_IK71Ed-Fi8rA7Di7ZA" target="_Mxr3MK71Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_vhHOMa72Ed-
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_vhHOMq72Ed-Fi8rA7Di7ZA" points="[-34, -33, 54, 54]$[-84, -83,
4, 4]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_wmaCUK72Ed-Fi8rA7Di7ZA"
id="(0.07894736842105263,0.5606060606060606)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_vhHOM672Ed-Fi8rA7Di7ZA" id="(0.5702479338842975,0.9)"/>
  </edges>
  <edges xmi:type="notation:Connector" xmi:id="_x78R4K72Ed-Fi8rA7Di7ZA"
type="4012" source="_Mxr3MK71Ed-Fi8rA7Di7ZA" target="_208qEK70Ed-
Fi8rA7Di7ZA"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_x78R4a72Ed-
```



```
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_x78R4q72Ed-Fi8rA7Di7ZA" points="[-19, -20, 59, 63]$[-64, -71,
14, 12]"/>
  <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_x8Fb0K72Ed-Fi8rA7Di7ZA"
id="(0.5757575757575758,0.03278688524590164)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_0WUcKk72Ed-Fi8rA7Di7ZA"
type="4011" source="_208qEK70Ed-Fi8rA7Di7ZA" target="_ML1YIK71Ed-
Fi8rA7Di7ZA">
  <styles xmi:type="notation:FontStyle" xmi:id="_0WUcKa72Ed-
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_0WUcKq72Ed-Fi8rA7Di7ZA" points="[-9, 31, 32, -124]$[-41, 155,
0, 0]"/>
  <targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_0WUcK672Ed-Fi8rA7Di7ZA" id="(0.18181818181818182,0.0)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_18sKgK72Ed-Fi8rA7Di7ZA"
type="4012" source="_ML1YIK71Ed-Fi8rA7Di7ZA" target="_Dis_IK71Ed-
Fi8rA7Di7ZA">
  <styles xmi:type="notation:FontStyle" xmi:id="_18sKga72Ed-
Fi8rA7Di7ZA"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_18sKgq72Ed-Fi8rA7Di7ZA" points="[21, 4, -136, -20]$[119, 38, -
38, 14]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_18sKg672Ed-Fi8rA7Di7ZA" id="(0.8264462809917356,0.6)"/>
</edges>
</notation:Diagram>
```



Λίστα 7. Το SUC μοντέλο σε XML μορφή (αρχείο SUC_example.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<notation:Diagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SUC="http://www.amcl.tuc.gr/aseme/metamodels/SUC"
xmlns:notation="http://www.eclipse.org/gmf/runtime/1.0.2/notation"
xmi:id="_qc5RQLdYEd-XdeI4awpGJg" type="SUC" name="SUC_sample.sucd"
measurementUnit="Pixel">
  <children xmi:type="notation:Node" xmi:id="_sQhyELdYEd-XdeI4awpGJg"
type="2009">
    <children xmi:type="notation:DecorationNode" xmi:id="_sQhyFLdYEd-
XdeI4awpGJg" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_sQhyFbdYEd-
XdeI4awpGJg" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_sQhyEbdYEd-
XdeI4awpGJg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_sQhyErdYEd-
XdeI4awpGJg"/>
    <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_sQhyE7dYEd-
XdeI4awpGJg" x="90" y="115" width="76"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_sypnsLdYEd-XdeI4awpGJg"
type="2009">
    <children xmi:type="notation:DecorationNode" xmi:id="_sypntLdYEd-
XdeI4awpGJg" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_sypntbdYEd-
XdeI4awpGJg" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_sypnsbdYEd-
XdeI4awpGJg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_sypnsrdYEd-
XdeI4awpGJg"/>
    <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.1"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_sypns7dYEd-
XdeI4awpGJg" x="85" y="220" width="61"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_tS-tkLdYEd-XdeI4awpGJg"
type="2009">
    <children xmi:type="notation:DecorationNode" xmi:id="_tS-tlLdYEd-
XdeI4awpGJg" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_tS-tlbdYEd-
XdeI4awpGJg" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_tS-tkbdYEd-
XdeI4awpGJg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_tS-tkrdYEd-
XdeI4awpGJg"/>
    <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.2"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_tS-tk7dYEd-
XdeI4awpGJg" x="80" y="330" width="71"/>
  </children>
  <children xmi:type="notation:Shape" xmi:id="_udVn4LdYEd-XdeI4awpGJg">
```



```
type="2008" bold="true" italic="true" fillColor="65535">
  <children xmi:type="notation:DecorationNode" xmi:id="_udVn4rdYEd-
XdeI4awpGJg" type="5010"/>
  <element xmi:type="SUC:HumanRole" href="SUC_sample.suc#//@roles.0"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_udVn4bdYEd-
XdeI4awpGJg" x="10" y="215"/>
</children>
<children xmi:type="notation:Node" xmi:id="_5c0XULdYEd-XdeI4awpGJg"
type="2009">
  <children xmi:type="notation:DecorationNode" xmi:id="_5c0XVLdYEd-
XdeI4awpGJg" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_5c0XVbdYEd-
XdeI4awpGJg" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_5c0XUbdYEd-
XdeI4awpGJg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_5c0XUrdYEd-
XdeI4awpGJg"/>
  <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.3"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_5c0XU7dYEd-
XdeI4awpGJg" x="225" y="50" width="111"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_7W3fELdYEd-XdeI4awpGJg"
type="2007" bold="true" fillColor="16776960">
  <children xmi:type="notation:DecorationNode" xmi:id="_7W3fErdYEd-
XdeI4awpGJg" type="5009"/>
  <element xmi:type="SUC:SystemRole"
href="SUC_sample.suc#//@roles.1"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_7W3fEbdYEd-
XdeI4awpGJg" x="175" y="190"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_Ej52YLdZEd-XdeI4awpGJg"
type="2007" bold="true" fillColor="8388863">
  <children xmi:type="notation:DecorationNode" xmi:id="_Ej52YrdZEd-
XdeI4awpGJg" type="5009"/>
  <element xmi:type="SUC:SystemRole"
href="SUC_sample.suc#//@roles.2"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Ej52YbdZEd-
XdeI4awpGJg" x="615" y="200"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_JXndALdZEd-XdeI4awpGJg"
type="2007" bold="true" fillColor="33023">
  <children xmi:type="notation:DecorationNode" xmi:id="_JXndArdZEd-
XdeI4awpGJg" type="5009"/>
  <element xmi:type="SUC:SystemRole"
href="SUC_sample.suc#//@roles.3"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_JXndAbdZEd-
XdeI4awpGJg" x="715" y="200" width="111"/>
</children>
<children xmi:type="notation:Node" xmi:id="_WAAg4LdZEd-XdeI4awpGJg"
type="2009">
  <children xmi:type="notation:DecorationNode" xmi:id="_WAAg5LdZEd-
XdeI4awpGJg" type="5011"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_WAAg5bdZEd-
```




```
XdeI4awpGJg" type="5012"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_WAAg4bdZEd-
XdeI4awpGJg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_WAAg4rdZEd-
XdeI4awpGJg"/>
  <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.4"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_WAAg47dZEd-
XdeI4awpGJg" x="365" y="70"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_WXP8MLdZEd-XdeI4awpGJg"
type="2009">
    <children xmi:type="notation:DecorationNode" xmi:id="_WXP8NLdZEd-
XdeI4awpGJg" type="5011"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_WXP8NbdZEd-
XdeI4awpGJg" type="5012"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_WXP8mbdZEd-
XdeI4awpGJg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_WXP8MrdZEd-
XdeI4awpGJg"/>
    <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.5"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_WXP8M7dZEd-
XdeI4awpGJg" x="505" y="120"/>
    </children>
    <children xmi:type="notation:Node" xmi:id="_Xl7V0LdZEd-XdeI4awpGJg"
type="2009">
      <children xmi:type="notation:DecorationNode" xmi:id="_Xl7V1LdZEd-
XdeI4awpGJg" type="5011"/>
      <children xmi:type="notation:DecorationNode" xmi:id="_Xl7V1bdZEd-
XdeI4awpGJg" type="5012"/>
      <styles xmi:type="notation:DescriptionStyle" xmi:id="_Xl7V0bdZEd-
XdeI4awpGJg"/>
      <styles xmi:type="notation:FontStyle" xmi:id="_Xl7V0rdZEd-
XdeI4awpGJg"/>
      <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.6"/>
      <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Xl7V07dZEd-
XdeI4awpGJg" x="405" y="325" width="95"/>
      </children>
      <children xmi:type="notation:Node" xmi:id="_Yj6oELdZEd-XdeI4awpGJg"
type="2009">
        <children xmi:type="notation:DecorationNode" xmi:id="_Yj6oFLdZEd-
XdeI4awpGJg" type="5011"/>
        <children xmi:type="notation:DecorationNode" xmi:id="_Yj6oFbdZEd-
XdeI4awpGJg" type="5012"/>
        <styles xmi:type="notation:DescriptionStyle" xmi:id="_Yj6oEbdZEd-
XdeI4awpGJg"/>
        <styles xmi:type="notation:FontStyle" xmi:id="_Yj6oErdZEd-
XdeI4awpGJg"/>
        <element xmi:type="SUC:UseCase"
href="SUC_sample.suc#//@usecases.7"/>
        <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Yj6oE7dZEd-
XdeI4awpGJg" x="670" y="50" width="87"/>
```




```
</children>
<children xmi:type="notation:Shape" xmi:id="_lrCZ4LdaEd-XdeI4awpGJg"
type="Text" description="include">
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_lrCZ4rdaEd-XdeI4awpGJg" type="DiagramName">
    <element xsi:nil="true"/>
  </children>
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_lrCZ47daEd-XdeI4awpGJg" type="Description">
    <element xsi:nil="true"/>
  </children>
  <element xsi:nil="true"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_lrCZ4bdaEd-
XdeI4awpGJg" x="150" y="85" width="66"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_oUw0ILdaEd-XdeI4awpGJg"
type="Text" description="include">
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_oUw0IrdEd-XdeI4awpGJg" type="DiagramName">
    <element xsi:nil="true"/>
  </children>
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_oUw0I7daEd-XdeI4awpGJg" type="Description">
    <element xsi:nil="true"/>
  </children>
  <element xsi:nil="true"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_oUw0IbdaEd-
XdeI4awpGJg" x="90" y="180" width="58"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_q7DiALdaEd-XdeI4awpGJg"
type="Text" description="include">
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_q7DiArdaEd-XdeI4awpGJg" type="DiagramName">
    <element xsi:nil="true"/>
  </children>
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_q7DiA7daEd-XdeI4awpGJg" type="Description">
    <element xsi:nil="true"/>
  </children>
  <element xsi:nil="true"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_q7DiAbdaEd-
XdeI4awpGJg" x="220" y="285" width="76"/>
</children>
<children xmi:type="notation:Shape" xmi:id="_uusLsLdaEd-XdeI4awpGJg"
type="Text" description="include">
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_uusLsrdaEd-XdeI4awpGJg" type="DiagramName">
    <element xsi:nil="true"/>
  </children>
  <children xmi:type="notation:BasicDecorationNode"
xmi:id="_uusLs7daEd-XdeI4awpGJg" type="Description">
    <element xsi:nil="true"/>
  </children>
  <element xsi:nil="true"/>
```



```
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_uusLsbdaEd-  
XdeI4awpGJg" x="335" y="195" width="68"/>  
</children>  
<children xmi:type="notation:Shape" xmi:id="_xEciILdaEd-XdeI4awpGJg"  
type="Text" description="include">  
  <children xmi:type="notation:BasicDecorationNode"  
xmi:id="_xEciIrdaEd-XdeI4awpGJg" type="DiagramName">  
    <element xsi:nil="true"/>  
  </children>  
  <children xmi:type="notation:BasicDecorationNode"  
xmi:id="_xEciI7daEd-XdeI4awpGJg" type="Description">  
    <element xsi:nil="true"/>  
  </children>  
  <element xsi:nil="true"/>  
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_xEciIbdaEd-  
XdeI4awpGJg" x="395" y="230" width="72"/>  
</children>  
<children xmi:type="notation:Shape" xmi:id="_zde2ELdaEd-XdeI4awpGJg"  
type="Text" description="include">  
  <children xmi:type="notation:BasicDecorationNode"  
xmi:id="_zde2ErdaEd-XdeI4awpGJg" type="DiagramName">  
    <element xsi:nil="true"/>  
  </children>  
  <children xmi:type="notation:BasicDecorationNode"  
xmi:id="_zde2E7daEd-XdeI4awpGJg" type="Description">  
    <element xsi:nil="true"/>  
  </children>  
  <element xsi:nil="true"/>  
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_zde2EbdaEd-  
XdeI4awpGJg" x="465" y="200" width="68"/>  
</children>  
<styles xmi:type="notation:DiagramStyle" xmi:id="_qc5RQbdYEd-  
XdeI4awpGJg"/>  
<element xmi:type="SUC:SUCmodel" href="SUC_sample.suc#/">  
  <edges xmi:type="notation:Connector" xmi:id="_pBRyYLdZEd-XdeI4awpGJg"  
type="4008" source="_Ej52YLdZEd-XdeI4awpGJg" target="_Yj6oELdZEd-  
XdeI4awpGJg">  
    <styles xmi:type="notation:FontStyle" xmi:id="_pBRyYbdZEd-  
XdeI4awpGJg"/>  
    <element xsi:nil="true"/>  
    <endpoints xmi:type="notation:RelativeEndpoints"  
xmi:id="_pBRyYrdZEd-XdeI4awpGJg" points="[10, -20, -59, 115]$" [68, -130,  
-1, 5]/>  
    <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_pBRyY7dZEd-XdeI4awpGJg" id="(0.425,0.875)"/>  
  </edges>  
  <edges xmi:type="notation:Connector" xmi:id="_pizyILdZEd-XdeI4awpGJg"  
type="4008" source="_JXndALdZEd-XdeI4awpGJg" target="_Yj6oELdZEd-  
XdeI4awpGJg">  
    <styles xmi:type="notation:FontStyle" xmi:id="_pizyIbdZEd-  
XdeI4awpGJg"/>  
    <element xsi:nil="true"/>  
    <endpoints xmi:type="notation:RelativeEndpoints"  
xmi:id="_pizyIrdZEd-XdeI4awpGJg" points="[-9, -20, 52, 115]$" [-64, -130,
```



```
-3, 5]"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_pizyI7dZEd-XdeI4awpGJg" id="(0.325,0.875)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_D28NQLdaEd-XdeI4awpGJg"
type="4008" source="_udVn4LdYEd-XdeI4awpGJg" target="_sypnsLdYEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_D28NQbdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_D28NQrdaEd-XdeI4awpGJg" points="[24, 1, -35, -3]${51, 3, -8, -
1]"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_D28NQ7daEd-XdeI4awpGJg" id="(0.13114754098360656,0.475)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_EnNzoLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_sQhyELdYEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_EnNzobdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_EnNzordaEd-XdeI4awpGJg" points="[-25, -20, 65, 52]${-75, -71,
15, 1]"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_EnNzo7daEd-XdeI4awpGJg" id="(0.9210526315789473,0.45)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_FmsToLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_5cOXULdYEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_FmsTobdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_FmsTordaEd-XdeI4awpGJg" points="[2, -4, -54, 104]${53, -104, -
3, 4]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_FmsTo7daEd-XdeI4awpGJg" id="(0.3106060606060606,0.1)"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_FmsTpLdaEd-XdeI4awpGJg" id="(0.42342342342342343,0.9)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_GiTAQLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_sypnsLdYEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_GiTAQbdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_GiTAQrdaEd-XdeI4awpGJg" points="[-26, 8, 50, -16]${-69, 25, 7,
1]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_GiTAQ7daEd-XdeI4awpGJg" id="(0.7348484848484849,0.9)"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
```



```
xmi:id="_GiTARLdaEd-XdeI4awpGJg" id="(0.8852459016393442,0.65)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_HmPwkLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_tS-tkLdYEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_HmPwkbdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_HmPwkrdaEd-XdeI4awpGJg" points="[-3, 5, 71, -120]$, [-39, 105,
35, -20]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_HmZhkLdaEd-XdeI4awpGJg" id="(0.6060606060606061,0.775)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_JaofULdaEd-XdeI4awpGJg"
type="4008" source="_Ej52YLdZEd-XdeI4awpGJg" target="_Xl7V0LdZEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_JaofUbdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_JaofUrdaEd-XdeI4awpGJg" points="[-29, 17, 119, -73]$, [-155, 83,
-7, -7]"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_JaofU7daEd-XdeI4awpGJg" id="(0.3684210526315789,0.175)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_L31hkLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_WAAg4LdZEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_L31hkLdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_L31hkrdaEd-XdeI4awpGJg" points="[7, -4, -158, 90]$, [149, -89, -
16, 5]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_L31hk7daEd-XdeI4awpGJg" id="(0.19696969696969696,0.2)"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_L31hlLdaEd-XdeI4awpGJg" id="(0.5813953488372093,0.9)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_NqH-kLdaEd-XdeI4awpGJg"
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_WXP8MLdZEd-
XdeI4awpGJg">
<styles xmi:type="notation:FontStyle" xmi:id="_NqH-kbdaEd-
XdeI4awpGJg"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_NqH-
krdaEd-XdeI4awpGJg" points="[66, -16, -209, 48]$, [264, -62, -11, 2]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_Od1TcLdaEd-XdeI4awpGJg" id="(0.23484848484848486,0.025)"/>
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_NqH-
k7daEd-XdeI4awpGJg" id="(0.15277777777777778,0.65)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_Q0MG0LdaEd-XdeI4awpGJg"
```



```
type="4008" source="_7W3fELdYEd-XdeI4awpGJg" target="_Xl7V0LdZEd-  
XdeI4awpGJg">  
  <styles xmi:type="notation:FontStyle" xmi:id="_Q0MG0bdaEd-  
XdeI4awpGJg"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_Q0MG0rdaEd-XdeI4awpGJg" points="[16, 6, -283, -104]$, [278, 118,  
-21, 8]"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_Q0MG07daEd-XdeI4awpGJg" id="(0.5075757575757576,0.925)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_Q0MG1LdaEd-XdeI4awpGJg" id="(0.11578947368421053,0.65)"/>  
  </edges>  
  <edges xmi:type="notation:Connector" xmi:id="_YgE0cLdaEd-XdeI4awpGJg"  
type="4009" source="_5c0XULdYEd-XdeI4awpGJg" target="_sQhyELdYEd-  
XdeI4awpGJg">  
    <styles xmi:type="notation:FontStyle" xmi:id="_YgE0cbdaEd-  
XdeI4awpGJg"/>  
    <element xsi:nil="true"/>  
    <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_YgE0crdaEd-XdeI4awpGJg" points="[-25, 15, 72, -46]$, [-84, 58,  
13, -3]"/>  
    <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_YgE0c7daEd-XdeI4awpGJg" id="(0.22522522522522523,0.5)"/>  
    <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_YgE0dLdaEd-XdeI4awpGJg" id="(0.5131578947368421,0.15)"/>  
    </edges>  
    <edges xmi:type="notation:Connector" xmi:id="_acNJ8LdaEd-XdeI4awpGJg"  
type="4009" source="_sypnsLdYEd-XdeI4awpGJg" target="_sQhyELdYEd-  
XdeI4awpGJg">  
      <styles xmi:type="notation:FontStyle" xmi:id="_acNJ8bdaEd-  
XdeI4awpGJg"/>  
      <element xsi:nil="true"/>  
      <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_acNJ8rdaEd-XdeI4awpGJg" points="[-2, -9, -19, 85]$, [-21, -80, -  
38, 14]"/>  
      <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_acWT4LdaEd-XdeI4awpGJg" id="(0.4262295081967213,0.225)"/>  
      </edges>  
      <edges xmi:type="notation:Connector" xmi:id="_b6hiELdaEd-XdeI4awpGJg"  
type="4009" source="_sypnsLdYEd-XdeI4awpGJg" target="_Xl7V0LdZEd-  
XdeI4awpGJg">  
        <styles xmi:type="notation:FontStyle" xmi:id="_b6hiEbdaEd-  
XdeI4awpGJg"/>  
        <element xsi:nil="true"/>  
        <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_b6hiErdaEd-XdeI4awpGJg" points="[31, 10, -273, -90]$, [290, 97, -  
14, -3]"/>  
        <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_d8A4sLdaEd-XdeI4awpGJg" id="(0.4262295081967213,0.225)"/>  
        <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_b6hiE7daEd-XdeI4awpGJg" id="(0.06315789473684211,0.7)"/>  
        </edges>  
        <edges xmi:type="notation:Connector" xmi:id="_heXlILdaEd-XdeI4awpGJg"
```



```
type="4009" source="_5c0XULdYEd-XdeI4awpGJg" target="_Xl7V0LdZEd-  
XdeI4awpGJg">  
  <styles xmi:type="notation:FontStyle" xmi:id="_heXlIbdaEd-  
XdeI4awpGJg"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_heXlI7daEd-XdeI4awpGJg" points="[14, 20, -161, -238]$, [175, 255,  
0, -3]"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_heXlI7daEd-XdeI4awpGJg" id="(0.5263157894736842,0.075)"/>  
  </edges>  
  <edges xmi:type="notation:Connector" xmi:id="_i36yALdaEd-XdeI4awpGJg"  
type="4009" source="_Xl7V0LdZEd-XdeI4awpGJg" target="_WAAg4LdZEd-  
XdeI4awpGJg">  
    <styles xmi:type="notation:FontStyle" xmi:id="_i36yAbdaEd-  
XdeI4awpGJg"/>  
    <element xsi:nil="true"/>  
    <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_i36yArdaEd-XdeI4awpGJg" points="[-1, -7, 30, 221]$, [-27, -222,  
4, 6]"/>  
    <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_i36yA7daEd-XdeI4awpGJg" id="(0.4631578947368421,0.175)"/>  
    <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_i4EjALdaEd-XdeI4awpGJg" id="(0.6162790697674418,0.85)"/>  
    </edges>  
    <edges xmi:type="notation:Connector" xmi:id="_jvD1YLdaEd-XdeI4awpGJg"  
type="4009" source="_Xl7V0LdZEd-XdeI4awpGJg" target="_WXP8MLdZEd-  
XdeI4awpGJg">  
      <styles xmi:type="notation:FontStyle" xmi:id="_jvD1YbdaEd-  
XdeI4awpGJg"/>  
      <element xsi:nil="true"/>  
      <bendpoints xmi:type="notation:RelativeBendpoints"  
xmi:id="_jvD1YrdaEd-XdeI4awpGJg" points="[7, -20, -60, 175]$, [53, -189, -  
14, 6]"/>  
      <sourceAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_kQIiILdaEd-XdeI4awpGJg" id="(0.5157894736842106,0.175)"/>  
      <targetAnchor xmi:type="notation:IdentityAnchor"  
xmi:id="_jvM_ULdaEd-XdeI4awpGJg" id="(0.19444444444444445,0.75)"/>  
      </edges>  
    </notation:Diagram>
```




Λίστα 8. Το AIP μοντέλο σε XML φορμά (αρχείο AIP_example.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<notation:Diagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:AIP="http://www.acml.tuc.gr/aseme/metamodels/AIP"
xmlns:notation="http://www.eclipse.org/gmf/runtime/1.0.2/notation"
xmi:id="_rXBaoK9qEd-vIpUUX-lAlw" type="AIP" name="AIP_sample.aipd"
measurementUnit="Pixel">
  <children xmi:type="notation:Node" xmi:id="_zRqV8K9qEd-vIpUUX-lAlw"
type="2004">
    <children xmi:type="notation:DecorationNode" xmi:id="_zR0G8K9qEd-
vIpUUX-lAlw" type="5010"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_zRqV8a9qEd-
vIpUUX-lAlw"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_zRqV8q9qEd-vIpUUX-
lAlw"/>
    <element xmi:type="AIP:Protocol"
href="AIP_sample.aip#//@protocols.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_zRqV869qEd-
vIpUUX-lAlw" x="165" y="35"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_3DgP4K9qEd-vIpUUX-lAlw"
type="2003">
    <children xmi:type="notation:DecorationNode" xmi:id="_3DgP5K9qEd-
vIpUUX-lAlw" type="5006"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_3DgP5a9qEd-
vIpUUX-lAlw" type="5007"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_3DgP5q9qEd-
vIpUUX-lAlw" type="5008"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_3DgP569qEd-
vIpUUX-lAlw" type="5009"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_3DgP4a9qEd-
vIpUUX-lAlw"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_3DgP4q9qEd-vIpUUX-
lAlw"/>
    <element xmi:type="AIP:Participant"
href="AIP_sample.aip#//@participants.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_3DgP469qEd-
vIpUUX-lAlw" x="15" y="175" width="152"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_-Yv_IK9qEd-vIpUUX-lAlw"
type="2003">
    <children xmi:type="notation:DecorationNode" xmi:id="_-Y5JEK9qEd-
vIpUUX-lAlw" type="5006"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_-Y5JEa9qEd-
vIpUUX-lAlw" type="5007"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_-Y5JEq9qEd-
vIpUUX-lAlw" type="5008"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_-Y5JE69qEd-
vIpUUX-lAlw" type="5009"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_-Yv_Ia9qEd-
vIpUUX-lAlw"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_-Yv_Iq9qEd-vIpUUX-
lAlw"/>
  </children>
</notation:Diagram>
```




```
<element xmi:type="AIP:Participant"
href="AIP_sample.aip#//@participants.1"/>
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_Yv_I69qEd-
vIpUUX-lAlw" x="270" y="170" width="146"/>
</children>
<styles xmi:type="notation:DiagramStyle" xmi:id="_rXBaoa9qEd-vIpUUX-
lAlw"/>
<element xmi:type="AIP:AIPmodel" href="AIP_sample.aip#/" />
<edges xmi:type="notation:Connector" xmi:id="_OtxFwk9rEd-vIpUUX-lAlw"
type="4002" source="_zRqV8K9qEd-vIpUUX-lAlw" target="_3DgP4K9qEd-vIpUUX-
lAlw">
<styles xmi:type="notation:FontStyle" xmi:id="_OtxFwa9rEd-vIpUUX-
lAlw"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_OtxFwq9rEd-vIpUUX-lAlw" points="[-24, 20, 122, -100]$[-146,
120, 0, 0]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_TeykwK9rEd-vIpUUX-lAlw" id="(0.4782608695652174,0.975)"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_Oup2ka9rEd-vIpUUX-lAlw" id="(0.5328947368421053,0.0)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_PkJ7MK9rEd-vIpUUX-lAlw"
type="4002" source="_zRqV8K9qEd-vIpUUX-lAlw" target="_Yv_IK9qEd-vIpUUX-
lAlw">
<styles xmi:type="notation:FontStyle" xmi:id="_PkJ7Ma9rEd-vIpUUX-
lAlw"/>
<element xsi:nil="true"/>
<bendpoints xmi:type="notation:RelativeBendpoints"
xmi:id="_PkJ7Mq9rEd-vIpUUX-lAlw" points="[7, 9, -85, -97]$" [91, 104, -1,
-2]"/>
<sourceAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_PkJ7M69rEd-vIpUUX-lAlw" id="(0.5217391304347826,0.975)"/>
<targetAnchor xmi:type="notation:IdentityAnchor"
xmi:id="_PkJ7NK9rEd-vIpUUX-lAlw"
id="(0.4794520547945205,0.02531645569620253)"/>
</edges>
</notation:Diagram>
```



Λίστα 9. Το SRM μοντέλο σε XML φορμά (αρχείο SRM_example.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<notation:Diagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SRM="http://www.amcl.tuc.gr/ase/metamodels/SRM"
xmlns:notation="http://www.eclipse.org/gmf/runtime/1.0.2/notation"
xmi:id="_Ivpesa-xEd-cwcT9j7mIwx" type="SRM" name="SRM_sample.srmd"
measurementUnit="Pixel">
  <children xmi:type="notation:Node" xmi:id="_MbF0oK-xEd-cwcT9j7mIwx"
type="2005">
    <children xmi:type="notation:DecorationNode" xmi:id="_MbF0pK-xEd-
cwcT9j7mIwx" type="5007"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_MbF0pa-xEd-
cwcT9j7mIwx" type="5008"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_MbF0oa-xEd-
cwcT9j7mIwx"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_MbF0oq-xEd-
cwcT9j7mIwx"/>
    <element xmi:type="SRM:Role" href="SRM_sample.srm#//@roles.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_MbF0o6-xEd-
cwcT9j7mIwx" x="285" y="210" width="216" height="99"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_aGjqsK-yEd-cwcT9j7mIwx"
type="2004">
    <children xmi:type="notation:DecorationNode" xmi:id="_aGjqtK-yEd-
cwcT9j7mIwx" type="5006"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_aGjqsa-yEd-
cwcT9j7mIwx"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_aGjqsq-yEd-
cwcT9j7mIwx"/>
    <element xmi:type="SRM:Capability"
href="SRM_sample.srm#//@capabilities.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_aGjqs6-yEd-
cwcT9j7mIwx" x="260" y="50"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_esG5gK-yEd-cwcT9j7mIwx"
type="2004">
    <children xmi:type="notation:DecorationNode" xmi:id="_esG5hK-yEd-
cwcT9j7mIwx" type="5006"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_esG5ga-yEd-
cwcT9j7mIwx"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_esG5gq-yEd-
cwcT9j7mIwx"/>
    <element xmi:type="SRM:Capability"
href="SRM_sample.srm#//@capabilities.1"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_esG5g6-yEd-
cwcT9j7mIwx" x="555" y="55"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_odEdQK-yEd-cwcT9j7mIwx"
type="2006">
    <children xmi:type="notation:DecorationNode" xmi:id="_odEdRK-yEd-
cwcT9j7mIwx" type="5009"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_odEdRa-yEd-
cwcT9j7mIwx" type="5010"/>
  </children>
</children>
</notation:Diagram>
```



```
<styles xmi:type="notation:DescriptionStyle" xmi:id="_odEdQa-yEd-  
cwcT9j7mIwx"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_odEdQq-yEd-  
cwcT9j7mIwx"/>  
<element xmi:type="SRM:Activity"  
href="SRM_sample.srm#//@activities.0"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_odEdQ6-yEd-  
cwcT9j7mIwx" x="5" y="130"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_qnftcK-yEd-cwcT9j7mIwx"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_qnftdK-yEd-  
cwcT9j7mIwx" type="5009"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_qnftda-yEd-  
cwcT9j7mIwx" type="5010"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_qnftca-yEd-  
cwcT9j7mIwx"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_qnftcq-yEd-  
cwcT9j7mIwx"/>  
<element xmi:type="SRM:Activity"  
href="SRM_sample.srm#//@activities.1"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_qnftc6-yEd-  
cwcT9j7mIwx" x="10" y="185"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_yFE5UK-yEd-cwcT9j7mIwx"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_yFE5VK-yEd-  
cwcT9j7mIwx" type="5009"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_yFE5Va-yEd-  
cwcT9j7mIwx" type="5010"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_yFE5Ua-yEd-  
cwcT9j7mIwx"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_yFE5Uq-yEd-  
cwcT9j7mIwx"/>  
<element xmi:type="SRM:Activity"  
href="SRM_sample.srm#//@activities.2"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_yFE5U6-yEd-  
cwcT9j7mIwx" x="35" y="240"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_40eJgK-yEd-cwcT9j7mIwx"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_40eJhK-yEd-  
cwcT9j7mIwx" type="5009"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_40eJha-yEd-  
cwcT9j7mIwx" type="5010"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_40eJga-yEd-  
cwcT9j7mIwx"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_40eJgq-yEd-  
cwcT9j7mIwx"/>  
<element xmi:type="SRM:Activity"  
href="SRM_sample.srm#//@activities.3"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_40eJg6-yEd-  
cwcT9j7mIwx" x="45" y="290"/>  
</children>
```



```
<children xmi:type="notation:Node" xmi:id="_eklkK-yEd-cwcT9j7mIwx"
type="2006">
  <children xmi:type="notation:DecorationNode" xmi:id="_ekllK-yEd-
cwcT9j7mIwx" type="5009"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_eklla-yEd-
cwcT9j7mIwx" type="5010"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_eklka-yEd-
cwcT9j7mIwx"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_eklkq-yEd-
cwcT9j7mIwx"/>
  <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.4"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_eklk6-yEd-
cwcT9j7mIwx" x="70" y="345"/>
</children>
<children xmi:type="notation:Node" xmi:id="_E5Fx8K-zEd-cwcT9j7mIwx"
type="2006">
  <children xmi:type="notation:DecorationNode" xmi:id="_E5Fx9K-zEd-
cwcT9j7mIwx" type="5009"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_E5Fx9a-zEd-
cwcT9j7mIwx" type="5010"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_E5Fx8a-zEd-
cwcT9j7mIwx"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_E5Fx8q-zEd-
cwcT9j7mIwx"/>
  <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.5"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_E5Fx86-zEd-
cwcT9j7mIwx" x="104" y="411"/>
</children>
<children xmi:type="notation:Node" xmi:id="_iKnyUK-zEd-cwcT9j7mIwx"
type="2006">
  <children xmi:type="notation:DecorationNode" xmi:id="_iKnyVK-zEd-
cwcT9j7mIwx" type="5009"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_iKnyVa-zEd-
cwcT9j7mIwx" type="5010"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_iKnyUa-zEd-
cwcT9j7mIwx"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_iKnyUq-zEd-
cwcT9j7mIwx"/>
  <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.6"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_iKnyU6-zEd-
cwcT9j7mIwx" x="650" y="125"/>
</children>
<children xmi:type="notation:Node" xmi:id="_mXI2EK-zEd-cwcT9j7mIwx"
type="2006">
  <children xmi:type="notation:DecorationNode" xmi:id="_mXI2FK-zEd-
cwcT9j7mIwx" type="5009"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_mXI2Fa-zEd-
cwcT9j7mIwx" type="5010"/>
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_mXI2Ea-zEd-
cwcT9j7mIwx"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_mXI2Eq-zEd-
```



```
cwcT9j7mIw"/>
  <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.7"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_mXI2E6-zEd-
cwcT9j7mIw" x="635" y="180"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_qn1PMK-zEd-cwcT9j7mIw"
type="2006">
    <children xmi:type="notation:DecorationNode" xmi:id="_qn1PNK-zEd-
cwcT9j7mIw" type="5009"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_qn1PNa-zEd-
cwcT9j7mIw" type="5010"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_qn1PMa-zEd-
cwcT9j7mIw"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_qn1PMq-zEd-
cwcT9j7mIw"/>
    <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.8"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_qn1PM6-zEd-
cwcT9j7mIw" x="610" y="240"/>
    </children>
    <children xmi:type="notation:Node" xmi:id="_s6AtMK-zEd-cwcT9j7mIw"
type="2006">
      <children xmi:type="notation:DecorationNode" xmi:id="_s6AtNK-zEd-
cwcT9j7mIw" type="5009"/>
      <children xmi:type="notation:DecorationNode" xmi:id="_s6AtNa-zEd-
cwcT9j7mIw" type="5010"/>
      <styles xmi:type="notation:DescriptionStyle" xmi:id="_s6AtMa-zEd-
cwcT9j7mIw"/>
      <styles xmi:type="notation:FontStyle" xmi:id="_s6AtMq-zEd-
cwcT9j7mIw"/>
      <element xmi:type="SRM:Activity"
href="SRM_sample.srm#/@activities.9"/>
      <layoutConstraint xmi:type="notation:Bounds" xmi:id="_s6AtM6-zEd-
cwcT9j7mIw" x="580" y="300"/>
      </children>
      <styles xmi:type="notation:DiagramStyle" xmi:id="_Ivpesq-xEd-
cwcT9j7mIw"/>
      <element xmi:type="SRM:SRMmodel" href="SRM_sample.srm#/">
      <edges xmi:type="notation:Connector" xmi:id="_RRbLIK-0Ed-cwcT9j7mIw"
type="4004" source="_MbF0oK-xEd-cwcT9j7mIw" target="_aGjqsK-yEd-
cwcT9j7mIw">
        <styles xmi:type="notation:FontStyle" xmi:id="_RRbLIa-0Ed-
cwcT9j7mIw"/>
        <element xsi:nil="true"/>
        <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_RRbLIq-
0Ed-cwcT9j7mIw" points="[-23, -49, 74, 140]$[-55, -172, 42, 17]"/>
        </edges>
        <edges xmi:type="notation:Connector" xmi:id="_TPyVQK-0Ed-cwcT9j7mIw"
type="4004" source="_MbF0oK-xEd-cwcT9j7mIw" target="_esG5gK-yEd-
cwcT9j7mIw">
          <styles xmi:type="notation:FontStyle" xmi:id="_TPyVQa-0Ed-
cwcT9j7mIw"/>
          <element xsi:nil="true"/>

```



```
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_TPyVQq-  
0Ed-cwCT9j7mIwx" points="[62, -49, -151, 118]${207, -164, -6, 3}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_T1x-QK-  
0Ed-cwCT9j7mIwx" id="(0.37037037037037035,0.020202020202020204)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_TP7fMK-  
0Ed-cwCT9j7mIwx" id="(0.3072289156626506,0.925)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_Ypyf0K-0Ed-cwCT9j7mIwx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIwx" target="_odEdQK-yEd-  
cwCT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_Ypyf0a-0Ed-  
cwCT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_Ypyf0q-  
0Ed-cwCT9j7mIwx" points="[-11, 20, 171, -51]${-11, 20, 171, -51}[-174,  
68, 8, -3]"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_Ypyf06-  
0Ed-cwCT9j7mIwx" id="(0.9344262295081968,0.275)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_ZgBkQK-0Ed-cwCT9j7mIwx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIwx" target="_qnftcK-yEd-  
cwCT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_ZgBkQa-0Ed-  
cwCT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_ZgBkQq-  
0Ed-cwCT9j7mIwx" points="[24, 12, 129, -99]${24, 12, 129, -99}[-100,  
107, 5, -4]"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ZgBkQ6-  
0Ed-cwCT9j7mIwx" id="(0.012048192771084338,0.7)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ZgBkRK-  
0Ed-cwCT9j7mIwx" id="(0.9358974358974359,0.1)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_aaJDIK-0Ed-cwCT9j7mIwx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIwx" target="_yFE5UK-yEd-  
cwCT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_aaJDIA-0Ed-  
cwCT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_aaJDIq-  
0Ed-cwCT9j7mIwx" points="[-3, 3, 113, -156]${-98, 154, 18, -5}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_aaJDI6-  
0Ed-cwCT9j7mIwx" id="(0.37349397590361444,0.925)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_aaJDJK-  
0Ed-cwCT9j7mIwx" id="(0.8860759493670886,0.15)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_bmuyEK-0Ed-cwCT9j7mIwx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIwx" target="_40eJgK-yEd-  
cwCT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_bmuyEa-0Ed-  
cwCT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_bmuyEq-  
0Ed-cwCT9j7mIwx" points="[-9, 20, 93, -207]${-88, 224, 14, -3}"/>
```




```
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_bmuyE6-  
0Ed-cwCT9j7mIxx" id="(0.9166666666666666,0.175)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_cbyKwK-0Ed-cwCT9j7mIxx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIxx" target="_-eklkK-yEd-  
cwCT9j7mIxx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_cbyKwa-0Ed-  
cwCT9j7mIxx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_cbyKwq-  
0Ed-cwCT9j7mIxx" points="[-3, 7, 75, -262]${-72, 267, 6, -2}"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_cbyKw6-  
0Ed-cwCT9j7mIxx" id="(0.40963855421686746,0.825)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_cbyKxK-  
0Ed-cwCT9j7mIxx" id="(0.9605263157894737,0.175)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_gJmgUK-0Ed-cwCT9j7mIxx"  
type="4006" source="_aGjqsK-yEd-cwCT9j7mIxx" target="_E5Fx8K-zEd-  
cwCT9j7mIxx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_gJmgUa-0Ed-  
cwCT9j7mIxx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_gJmgUq-  
0Ed-cwCT9j7mIxx" points="[-4, 20, 69, -328]${-65, 346, 8, -2}"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_gJmgU6-  
0Ed-cwCT9j7mIxx" id="(0.9393939393939394,0.175)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_jLKz4K-0Ed-cwCT9j7mIxx"  
type="4006" source="_esG5gK-yEd-cwCT9j7mIxx" target="_iKnyUK-zEd-  
cwCT9j7mIxx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_jLKz4a-0Ed-  
cwCT9j7mIxx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_jLKz4q-  
0Ed-cwCT9j7mIxx" points="[14, 11, -51, -39]${60, 49, -5, -1}"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_jLUk4K-  
0Ed-cwCT9j7mIxx" id="(0.21084337349397592,0.725)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_jLUk4a-  
0Ed-cwCT9j7mIxx" id="(0.028089887640449437,0.225)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_j7dBUK-0Ed-cwCT9j7mIxx"  
type="4006" source="_esG5gK-yEd-cwCT9j7mIxx" target="_mXI2EK-zEd-  
cwCT9j7mIxx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_j7dBUa-0Ed-  
cwCT9j7mIxx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_j7dBUq-  
0Ed-cwCT9j7mIxx" points="[3, 7, -38, -88]${36, 94, -5, -1}"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_j7dBU6-  
0Ed-cwCT9j7mIxx" id="(0.26506024096385544,0.825)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_j7dBVK-  
0Ed-cwCT9j7mIxx" id="(0.027777777777777776,0.075)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_mRNxwK-0Ed-cwCT9j7mIxx"
```




```
type="4006" source="_esG5gK-yEd-cwcT9j7mIwx" target="_qn1PMK-zEd-  
cwcT9j7mIwx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_mRNXwa-0Ed-  
cwcT9j7mIwx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_mRNXwq-  
0Ed-cwcT9j7mIwx" points="[0, 4, -19, -150]$, [8, 151, -11, -3]"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_mRNXw6-  
0Ed-cwcT9j7mIwx" id="(0.28313253012048195,0.9)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_mRNxxK-  
0Ed-cwcT9j7mIwx" id="(0.07051282051282051,0.125)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_nM90UK-0Ed-cwcT9j7mIwx"  
type="4006" source="_esG5gK-yEd-cwcT9j7mIwx" target="_s6AtMK-zEd-  
cwcT9j7mIwx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_nM90Ua-0Ed-  
cwcT9j7mIwx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_nM90Uq-  
0Ed-cwcT9j7mIwx" points="[-1, 6, 16, -211]$, [-22, 216, -5, -1]"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_nM90U6-  
0Ed-cwcT9j7mIwx" id="(0.28313253012048195,0.85)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_nM90VK-  
0Ed-cwcT9j7mIwx" id="(0.02976190476190476,0.15)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_ve_9AK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_odEdQK-yEd-  
cwcT9j7mIwx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_ve_9Aa-0Ed-  
cwcT9j7mIwx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_ve_9Aq-  
0Ed-cwcT9j7mIwx" points="[-25, -15, 166, 97]$, [-183, -114, 8, -2]"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ve_9A6-  
0Ed-cwcT9j7mIwx" id="(0.11574074074074074,0.45454545454545453)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ve_9BK-  
0Ed-cwcT9j7mIwx" id="(0.9344262295081968,0.325)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_wvwaQK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_qnftcK-yEd-  
cwcT9j7mIwx">  
  <styles xmi:type="notation:FontStyle" xmi:id="_wvwaQa-0Ed-  
cwcT9j7mIwx"/>  
  <element xsi:nil="true"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_wvwaQq-  
0Ed-cwcT9j7mIwx" points="[-17, -7, 123, 44]$, [-136, -52, 4, -1]"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_wv6LQK-  
0Ed-cwcT9j7mIwx" id="(0.0787037037037037,0.4444444444444444)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_wv6LQa-  
0Ed-cwcT9j7mIwx" id="(0.9743589743589743,0.45)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_xsjZSK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_yFE5UK-yEd-  
cwcT9j7mIwx">
```



```
<styles xmi:type="notation:FontStyle" xmi:id="_xsjZsa-0Ed-  
cwcT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_xsjZsq-  
0Ed-cwcT9j7mIwx" points="[-8, 1, 98, 0]${-100, 0, 6, -1}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_xstKsK-  
0Ed-cwcT9j7mIwx" id="(0.037037037037037035,0.41414141414141414)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_xstKsa-  
0Ed-cwcT9j7mIwx" id="(0.9620253164556962,0.3)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_ygHkoK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_-eklkK-yEd-  
cwcT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_ygQukK-0Ed-  
cwcT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_ygQuka-  
0Ed-cwcT9j7mIwx" points="[-16, 0, 69, -95]${-16, 0, 69, -95}[-81, 89,  
4, -6]"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ygQukq-  
0Ed-cwcT9j7mIwx" id="(0.07407407407407407,0.46464646464646464)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_ygQuk6-  
0Ed-cwcT9j7mIwx" id="(0.9605263157894737,0.15)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_zy9skK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_40eJgK-yEd-  
cwcT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_zy9ska-0Ed-  
cwcT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_zy9skq-  
0Ed-cwcT9j7mIwx" points="[-9, 4, 75, -41]${-81, 44, 3, -1}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_zy9sk6-  
0Ed-cwcT9j7mIwx" id="(0.041666666666666664,0.43434343434343436)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_zy9slK-  
0Ed-cwcT9j7mIwx" id="(0.9821428571428571,0.2)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_000yMK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_E5Fx8K-zEd-  
cwcT9j7mIwx">  
<styles xmi:type="notation:FontStyle" xmi:id="_000yMa-0Ed-  
cwcT9j7mIwx"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_000yMq-  
0Ed-cwcT9j7mIwx" points="[-13, 0, 53, -168]${-13, 0, 53, -168}[-62,  
155, 4, -13]"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_000yM6-  
0Ed-cwcT9j7mIwx" id="(0.06018518518518518,0.48484848484848486)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_000yNK-  
0Ed-cwcT9j7mIwx" id="(0.9696969696969697,0.375)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_5hFtAK-0Ed-cwcT9j7mIwx"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIwx" target="_iKnyUK-zEd-  
cwcT9j7mIwx">
```



```
<styles xmi:type="notation:FontStyle" xmi:id="_5hFtAa-0Ed-  
cwcT9j7mIxw"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_5hFtAq-  
0Ed-cwcT9j7mIxw" points="[108, -47, -154, 67]${257, -114, -5, 0}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_9mc_4K-  
0Ed-cwcT9j7mIxw" id="(0.9490740740740741,0.5353535353535354)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_5hFtA6-  
0Ed-cwcT9j7mIxw" id="(0.028089887640449437,0.5)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_6GfgIK-0Ed-cwcT9j7mIxw"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIxw" target="_mXI2EK-zEd-  
cwcT9j7mIxw">  
<styles xmi:type="notation:FontStyle" xmi:id="_6GfgIa-0Ed-  
cwcT9j7mIxw"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_6GfgIq-  
0Ed-cwcT9j7mIxw" points="[38, -12, -137, 40]${172, -53, -3, -1}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_6GfgI6-  
0Ed-cwcT9j7mIxw" id="(0.9166666666666666,0.5252525252525253)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_6GfgJK-  
0Ed-cwcT9j7mIxw" id="(0.016666666666666666,0.45)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_66MN8K-0Ed-cwcT9j7mIxw"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIxw" target="_qn1PMK-zEd-  
cwcT9j7mIxw">  
<styles xmi:type="notation:FontStyle" xmi:id="_66MN8a-0Ed-  
cwcT9j7mIxw"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_66MN8q-  
0Ed-cwcT9j7mIxw" points="[108, -2, -117, 0]${217, -3, -8, -1}"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_66MN86-  
0Ed-cwcT9j7mIxw" id="(0.05128205128205128,0.425)"/>  
</edges>  
<edges xmi:type="notation:Connector" xmi:id="_7i3-kK-0Ed-cwcT9j7mIxw"  
type="4005" source="_MbF0oK-xEd-cwcT9j7mIxw" target="_s6AtMK-zEd-  
cwcT9j7mIxw">  
<styles xmi:type="notation:FontStyle" xmi:id="_7i3-ka-0Ed-  
cwcT9j7mIxw"/>  
<element xsi:nil="true"/>  
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_7i3-kq-  
0Ed-cwcT9j7mIxw" points="[42, 21, -89, -46]${121, 67, -10, 0}"/>  
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_7jBvkk-  
0Ed-cwcT9j7mIxw" id="(0.9861111111111112,0.43434343434343436)"/>  
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_7jBvka-  
0Ed-cwcT9j7mIxw" id="(0.05952380952380952,0.575)"/>  
</edges>  
</notation:Diagram>
```



Λίστα 10. Το IAC μοντέλο σε XML φορμά (αρχείο IAC_example.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<notation:Diagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:IAC="http://www.amcl.tuc.gr/ase/metamodels/IAC"
xmlns:notation="http://www.eclipse.org/gmf/runtime/1.0.2/notation"
xmi:id="_AGq_Ea-jEd-of-gKNbzXRg" type="IAC" name="IAC_sample.iacd"
measurementUnit="Pixel">
  <children xmi:type="notation:Node" xmi:id="_DB-5oK-jEd-of-gKNbzXRg"
type="2006">
    <children xmi:type="notation:DecorationNode" xmi:id="_DCIqoK-jEd-of-
gKNbzXRg" type="5015"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_DCIqoa-jEd-of-
gKNbzXRg" type="5016"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_DCIqoq-jEd-of-
gKNbzXRg" type="5017"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_DCIqo6-jEd-of-
gKNbzXRg" type="5018"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_DB-5oa-jEd-of-
gKNbzXRg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_DB-5oq-jEd-of-
gKNbzXRg"/>
    <element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.0"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_DB-5o6-jEd-of-
gKNbzXRg" x="355" y="10"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_Fa3ckK-jEd-of-gKNbzXRg"
type="2006">
    <children xmi:type="notation:DecorationNode" xmi:id="_Fa3clK-jEd-of-
gKNbzXRg" type="5015"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_Fa3cla-jEd-of-
gKNbzXRg" type="5016"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_Fa3clq-jEd-of-
gKNbzXRg" type="5017"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_Fa3cl6-jEd-of-
gKNbzXRg" type="5018"/>
    <styles xmi:type="notation:DescriptionStyle" xmi:id="_Fa3cka-jEd-of-
gKNbzXRg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_Fa3ckq-jEd-of-
gKNbzXRg"/>
    <element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.1"/>
    <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Fa3ck6-jEd-of-
gKNbzXRg" x="19" y="124"/>
  </children>
  <children xmi:type="notation:Node" xmi:id="_GMyowK-jEd-of-gKNbzXRg"
type="2006">
    <children xmi:type="notation:DecorationNode" xmi:id="_GMyoxK-jEd-of-
gKNbzXRg" type="5015"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_GMyoxa-jEd-of-
gKNbzXRg" type="5016"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_GMyoxq-jEd-of-
gKNbzXRg" type="5017"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_GMyox6-jEd-of-
gKNbzXRg" type="5018"/>
  </children>
</children>
</notation:Diagram>
```



```
<styles xmi:type="notation:DescriptionStyle" xmi:id="_GMyowa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_GMyowq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.2"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_GMyow6-jEd-of-  
gKNbzXRg" x="135" y="125" width="111"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_G4AI8K-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_G4AI9K-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_G4AI9a-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_G4AI9q-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_G4AI96-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_G4AI8a-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_G4AI8q-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.3"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_G4AI86-jEd-of-  
gKNbzXRg" x="355" y="140" width="101"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_HZYXsK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_HZYXtK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_HZYXta-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_HZYXtq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_HZYXt6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_HZYXsa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_HZYXsq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.4"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_HZYXs6-jEd-of-  
gKNbzXRg" x="555" y="135"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_IBncYK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_IBncZK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_IBncZa-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_IBncZq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_IBncZ6-jEd-of-  
gKNbzXRg" type="5018"/>
```



```
<styles xmi:type="notation:DescriptionStyle" xmi:id="_IBncYa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_IBncYq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.5"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_IBncY6-jEd-of-  
gKNbzXRg" x="705" y="135" width="51"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_KInlIK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_KInlJK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_KInlJa-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_KInlJq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_KInlJ6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_KInlIa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_KInlIq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.6"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_KInlI6-jEd-of-  
gKNbzXRg" x="360" y="255" width="86"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_LRLvSK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_LRLvtK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_LRLvta-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_LRLvtq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_LRLvt6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_LRLvsa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_LRLvsq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.7"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_LRLvs6-jEd-of-  
gKNbzXRg" x="185" y="260"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_L4hcgK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_L4hchK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_L4hcha-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_L4hchq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_L4hch6-jEd-of-  
gKNbzXRg" type="5018"/>
```




```
<styles xmi:type="notation:DescriptionStyle" xmi:id="_L4hcga-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_L4hcgq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.8"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_L4hcg6-jEd-of-  
gKNbzXRg" x="515" y="255" width="51"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_e7cT0K-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_e7cT1K-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_e7cT1a-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_e7cT1q-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_e7cT16-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_e7cT0a-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_e7cT0q-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.9"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_e7cT06-jEd-of-  
gKNbzXRg" x="15" y="450"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_foltsK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_folttK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_foltta-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_folttq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_foltt6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_foltsa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_foltsq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.10"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_folts6-jEd-of-  
gKNbzXRg" x="146" y="459"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_gTWh8K-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_gTWh9K-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_gTWh9a-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_gTWh9q-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_gTWh96-jEd-of-  
gKNbzXRg" type="5018"/>
```




```
<styles xmi:type="notation:DescriptionStyle" xmi:id="_gTWh8a-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_gTWh8q-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.11"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_gTWh86-jEd-of-  
gKNbzXRg" x="340" y="465"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_g4wVEK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_g4wVFK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_g4wVFq-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_g4wVFq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_g4wVF6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_g4wVEa-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_g4wVEq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.12"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_g4wVE6-jEd-of-  
gKNbzXRg" x="475" y="465"/>  
</children>  
<children xmi:type="notation:Node" xmi:id="_hiLskK-jEd-of-gKNbzXRg"  
type="2006">  
<children xmi:type="notation:DecorationNode" xmi:id="_hiLslK-jEd-of-  
gKNbzXRg" type="5015"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_hiLsla-jEd-of-  
gKNbzXRg" type="5016"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_hiLslq-jEd-of-  
gKNbzXRg" type="5017"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_hiLsl6-jEd-of-  
gKNbzXRg" type="5018"/>  
<styles xmi:type="notation:DescriptionStyle" xmi:id="_hiLska-jEd-of-  
gKNbzXRg"/>  
<styles xmi:type="notation:FontStyle" xmi:id="_hiLskq-jEd-of-  
gKNbzXRg"/>  
<element xmi:type="IAC:Node" href="IAC_sample.iac#//@nodes.13"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_hiLsk6-jEd-of-  
gKNbzXRg" x="775" y="465"/>  
</children>  
<children xmi:type="notation:Shape" xmi:id="_H1qVYLaFE-d-V38bkejuuLg"  
type="2005">  
<children xmi:type="notation:DecorationNode" xmi:id="_H10GYLaFE-d-  
V38bkejuuLg" type="5013"/>  
<children xmi:type="notation:DecorationNode" xmi:id="_H10GYbaFE-d-  
V38bkejuuLg" type="5014"/>  
<element xmi:type="IAC:Variable"  
href="IAC_sample.iac#//@variables.0"/>  
<layoutConstraint xmi:type="notation:Bounds" xmi:id="_H1qVYbaFE-d-  
V38bkejuuLg" x="20" y="47" width="76"/>
```



```
</children>
<children xmi:type="notation:Shape" xmi:id="_Q1mE0LaFEd-V38bkejuuLg"
type="2005">
  <children xmi:type="notation:DecorationNode" xmi:id="_Q1mE0raFEd-
V38bkejuuLg" type="5013"/>
  <children xmi:type="notation:DecorationNode" xmi:id="_Q1mE07aFEd-
V38bkejuuLg" type="5014"/>
  <element xmi:type="IAC:Variable"
href="IAC_sample.iac#//@variables.1"/>
  <layoutConstraint xmi:type="notation:Bounds" xmi:id="_Q1mE0baFEd-
V38bkejuuLg" x="110" y="20" width="102" height="39"/>
</children>
<styles xmi:type="notation:DiagramStyle" xmi:id="_AGqEq-jEd-of-
gKNbzXRg"/>
<element xmi:type="IAC:Model" href="IAC_sample.iac#/" />
<edges xmi:type="notation:Connector" xmi:id="_j3WNIK-jEd-of-gKNbzXRg"
type="4008" source="_DB-5oK-jEd-of-gKNbzXRg" target="_Fa3ckK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_j3WNIa-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_j3WNIq-
jEd-of-gKNbzXRg" points="[-51, 5, 371, -37]$$[-402, 40, 20, -2]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_FNIlwK-
nEd-of-gKNbzXRg" id="(0.72,0.9367088607594937)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_j5uywK-
jEd-of-gKNbzXRg" id="(0.017543859649122806,0.05063291139240506)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_lBtHcK-jEd-of-gKNbzXRg"
type="4008" source="_DB-5oK-jEd-of-gKNbzXRg" target="_GMyowK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_lBtHca-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_lBtHcq-
jEd-of-gKNbzXRg" points="[-22, 15, 134, -90]$$[-131, 88, 25, -17]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_lBtHc6-
jEd-of-gKNbzXRg" id="(0.5733333333333334,0.8734177215189873)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_zqoYUK-
mEd-of-gKNbzXRg" id="(0.7747747747747747,0.43037974683544306)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_l0EYgK-jEd-of-gKNbzXRg"
type="4008" source="_DB-5oK-jEd-of-gKNbzXRg" target="_G4AI8K-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_l0EYga-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_l0EYgq-
jEd-of-gKNbzXRg" points="[-1, 40, 0, -51]$$[-1, 91, 0, 0]"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_l0Nick-
jEd-of-gKNbzXRg" id="(0.475,0.0)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_mWMOIK-jEd-of-gKNbzXRg"
type="4008" source="_DB-5oK-jEd-of-gKNbzXRg" target="_HZYXsK-jEd-of-
```



```
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_mWMOIa-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_mWMOIq-
jEd-of-gKNbzXRg" points="[20, 10, -173, -91]${188, 86, -5, -15}"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_001hoK-
mEd-of-gKNbzXRg" id="(0.20666666666666667,0.8860759493670886)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_mWMOI6-
jEd-of-gKNbzXRg" id="(0.45,0.05063291139240506)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_nkks0K-jEd-of-gKNbzXRg"
type="4008" source="_DB-5oK-jEd-of-gKNbzXRg" target="_IBncYK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_nkks0a-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_nkks0q-
jEd-of-gKNbzXRg" points="[75, 22, -222, -66]${290, 86, -7, -2}"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_2BSGoK-
mEd-of-gKNbzXRg" id="(0.08,0.8734177215189873)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_nkt2wK-
jEd-of-gKNbzXRg" id="(0.525,0.02531645569620253)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_o5zaYK-jEd-of-gKNbzXRg"
type="4008" source="_G4AI8K-jEd-of-gKNbzXRg" target="_KInlIK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_o5zaYa-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_o5zaYq-
jEd-of-gKNbzXRg" points="[0, 40, 0, -44]${-1, 81, -1, -3}"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_o5zaY6-
jEd-of-gKNbzXRg" id="(0.375,0.0379746835443038)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_poS4EK-jEd-of-gKNbzXRg"
type="4008" source="_KInlIK-jEd-of-gKNbzXRg" target="_gTWh8K-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_poS4Ea-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_poS4Eq-
jEd-of-gKNbzXRg" points="[-2, 40, 5, -125]${-6, 154, 1, -11}"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_poS4E6-
jEd-of-gKNbzXRg" id="(0.625,0.13924050632911392)"/>
</edges>
<edges xmi:type="notation:Connector" xmi:id="_rUetEK-jEd-of-gKNbzXRg"
type="4008" source="_G4AI8K-jEd-of-gKNbzXRg" target="_LRLvsK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_rUetEa-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_rUetEq-
jEd-of-gKNbzXRg" points="[29, 36, 181, -80]${30, 37, 182, -79}[-123,
```



```
104, 29, -12]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_3bR_cK-
mEd-of-gKNbzXRg" id="(0.09900990099009901,0.5443037974683544)"/>
</edges>
  <edges xmi:type="notation:Connector" xmi:id="_SB7o8K-jEd-of-gKNbzXRg"
type="4008" source="_G4AI8K-jEd-of-gKNbzXRg" target="_L4hcgK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_SB7o8a-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_SB7o8q-
jEd-of-gKNbzXRg" points="[-46, 45, -121, -62]$[-47, 46, -122, -61]$, [68,
103, -7, -4]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_414VMK-
mEd-of-gKNbzXRg" id="(0.9108910891089109,0.43037974683544306)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_sCEy4K-
jEd-of-gKNbzXRg" id="(0.15,0.3291139240506329)"/>
</edges>
  <edges xmi:type="notation:Connector" xmi:id="_ta4_8K-jEd-of-gKNbzXRg"
type="4008" source="_KInlIK-jEd-of-gKNbzXRg" target="_e7cT0K-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_ta4_8a-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_ta4_8q-
jEd-of-gKNbzXRg" points="[-20, 11, 325, -179]$[-325, 165, 20, -25]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_5bu0QK-
mEd-of-gKNbzXRg" id="(0.5930232558139535,0.9113924050632911)"/>
</edges>
  <edges xmi:type="notation:Connector" xmi:id="_uQYdgK-jEd-of-gKNbzXRg"
type="4008" source="_KInlIK-jEd-of-gKNbzXRg" target="_foltsK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_uQYdga-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_uQYdgq-
jEd-of-gKNbzXRg" points="[-7, 6, 182, -170]$[-181, 167, 8, -9]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_uQYdg6-
jEd-of-gKNbzXRg" id="(0.5,0.9367088607594937)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_-x3LcK-
mEd-of-gKNbzXRg" id="(0.5462962962962963,0.08860759493670886)"/>
</edges>
  <edges xmi:type="notation:Connector" xmi:id="_vBQg0K-jEd-of-gKNbzXRg"
type="4008" source="_KInlIK-jEd-of-gKNbzXRg" target="_g4wVEK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_vBQg0a-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_vBQg0q-
jEd-of-gKNbzXRg" points="[20, 24, -116, -142]$, [135, 164, -1, -2]"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_9W96wK-
mEd-of-gKNbzXRg" id="(0.43023255813953487,0.9113924050632911)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_vBQg06-
jEd-of-gKNbzXRg" id="(0.475,0.02531645569620253)"/>
```



```
</edges>
<edges xmi:type="notation:Connector" xmi:id="_vs6s8K-jEd-of-gKNbzXRg"
type="4008" source="_KInlIK-jEd-of-gKNbzXRg" target="_hiLskK-jEd-of-
gKNbzXRg">
  <styles xmi:type="notation:FontStyle" xmi:id="_vs6s8a-jEd-of-
gKNbzXRg"/>
  <element xsi:nil="true"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_vs6s8q-
jEd-of-gKNbzXRg" points="[20, 10, -357, -172]${371, 170, -6, -12}"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_AvCCwK-
nEd-of-gKNbzXRg" id="(0.1511627906976744,0.8227848101265823)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_vs6s86-
jEd-of-gKNbzXRg" id="(0.9655172413793104,0.13924050632911392)"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_eKoQAK-lEd-of-gKNbzXRg"
type="4007" source="_Fa3ckK-jEd-of-gKNbzXRg" target="_GMyowK-jEd-of-
gKNbzXRg">
  <children xmi:type="notation:DecorationNode" xmi:id="_eKoQBK-lEd-of-
gKNbzXRg" visible="false" type="6005">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_eKoQB-
lEd-of-gKNbzXRg" y="40"/>
  </children>
  <children xmi:type="notation:DecorationNode" xmi:id="_eKoQBq-lEd-of-
gKNbzXRg" visible="false" type="6006">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_eKoQB6-
lEd-of-gKNbzXRg" y="60"/>
  </children>
  <styles xmi:type="notation:RoutingStyle" xmi:id="_eKoQAa-lEd-of-
gKNbzXRg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_eKoQAq-lEd-of-
gKNbzXRg"/>
  <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.0"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_eKoQA6-
lEd-of-gKNbzXRg" points="[29, 1, -71, 0]${88, 0, -12, -1}"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_eL0i0K-
lEd-of-gKNbzXRg" id="(0.10810810810810811,0.4936708860759494)"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_fmtzgK-lEd-of-gKNbzXRg"
type="4007" source="_GMyowK-jEd-of-gKNbzXRg" target="_G4AI8K-jEd-of-
gKNbzXRg">
  <children xmi:type="notation:DecorationNode" xmi:id="_fmtzhK-lEd-of-
gKNbzXRg" visible="false" type="6005">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_fmtzha-
lEd-of-gKNbzXRg" y="40"/>
  </children>
  <children xmi:type="notation:DecorationNode" xmi:id="_fmtzhq-lEd-of-
gKNbzXRg" visible="false" type="6006">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_fmtzh6-
lEd-of-gKNbzXRg" y="60"/>
  </children>
  <styles xmi:type="notation:RoutingStyle" xmi:id="_fmtzga-lEd-of-
gKNbzXRg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_fmtzgq-lEd-of-
```




```
gKNbzXRg"/>
  <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.1"/>
    <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_fmtzg6-
lEd-of-gKNbzXRg" points="[56, 3, -127, -7]${165, 6, -18, -4}"/>
      <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_fmtziK-
lEd-of-gKNbzXRg" id="(0.1782178217821782,0.43037974683544306)"/>
    </edges>
    <edges xmi:type="notation:Edge" xmi:id="_gtDJcK-lEd-of-gKNbzXRg"
type="4007" source="_G4AI8K-jEd-of-gKNbzXRg" target="_HZYXsK-jEd-of-
gKNbzXRg"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_gtDJdK-lEd-of-
gKNbzXRg" visible="false" type="6005">
      <layoutConstraint xmi:type="notation:Location" xmi:id="_gtDJda-
lEd-of-gKNbzXRg" y="40"/>
    </children>
    <children xmi:type="notation:DecorationNode" xmi:id="_gtDJdq-lEd-of-
gKNbzXRg" visible="false" type="6006">
      <layoutConstraint xmi:type="notation:Location" xmi:id="_gtDJd6-
lEd-of-gKNbzXRg" y="60"/>
    </children>
    <styles xmi:type="notation:RoutingStyle" xmi:id="_gtDJca-lEd-of-
gKNbzXRg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_gtDJcq-lEd-of-
gKNbzXRg"/>
    <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.2"/>
      <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_gtDJc6-
lEd-of-gKNbzXRg" points="[51, -1, -119, 2]${150, -3, -20, 0}"/>
      <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_gtM6cK-
lEd-of-gKNbzXRg" id="(0.18181818181818182,0.5189873417721519)"/>
    </edges>
    <edges xmi:type="notation:Edge" xmi:id="_htIGYK-lEd-of-gKNbzXRg"
type="4007" source="_HZYXsK-jEd-of-gKNbzXRg" target="_IBncYK-jEd-of-
gKNbzXRg"/>
    <children xmi:type="notation:DecorationNode" xmi:id="_htIGZK-lEd-of-
gKNbzXRg" visible="false" type="6005">
      <layoutConstraint xmi:type="notation:Location" xmi:id="_htIGZa-
lEd-of-gKNbzXRg" y="40"/>
    </children>
    <children xmi:type="notation:DecorationNode" xmi:id="_htIGZq-lEd-of-
gKNbzXRg" visible="false" type="6006">
      <layoutConstraint xmi:type="notation:Location" xmi:id="_htIGZ6-
lEd-of-gKNbzXRg" y="60"/>
    </children>
    <styles xmi:type="notation:RoutingStyle" xmi:id="_htIGYa-lEd-of-
gKNbzXRg"/>
    <styles xmi:type="notation:FontStyle" xmi:id="_htIGYq-lEd-of-
gKNbzXRg"/>
    <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.3"/>
      <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_htIGY6-
lEd-of-gKNbzXRg" points="[55, -1, -42, 0]${95, -2, -2, -1}"/>
      <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_htIGaK-
```



```
lEd-of-gKNbzXRg" id="(0.047619047619047616,0.4810126582278481)"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_k0zY8K-lEd-of-gKNbzXRg"
type="4007" source="_LRLvsK-jEd-of-gKNbzXRg" target="_KInlIK-jEd-of-
gKNbzXRg">
  <children xmi:type="notation:DecorationNode" xmi:id="_k0zY9K-lEd-of-
gKNbzXRg" visible="false" type="6005">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_k0zY9a-
lEd-of-gKNbzXRg" y="40"/>
  </children>
  <children xmi:type="notation:DecorationNode" xmi:id="_k0zY9q-lEd-of-
gKNbzXRg" visible="false" type="6006">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_k0zY96-
lEd-of-gKNbzXRg" y="60"/>
  </children>
  <styles xmi:type="notation:RoutingStyle" xmi:id="_k0zY8a-lEd-of-
gKNbzXRg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_k0zY8q-lEd-of-
gKNbzXRg"/>
  <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.4"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_k0zY86-
lEd-of-gKNbzXRg" points="[29, -1, -161, 4]${147, -13, -43, -8}"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_qjqpQK-lEd-of-gKNbzXRg"
type="4007" source="_KInlIK-jEd-of-gKNbzXRg" target="_KInlIK-jEd-of-
gKNbzXRg">
  <children xmi:type="notation:DecorationNode" xmi:id="_qj0aQK-lEd-of-
gKNbzXRg" type="6005">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_qj0aQa-
lEd-of-gKNbzXRg" x="12" y="-7"/>
  </children>
  <children xmi:type="notation:DecorationNode" xmi:id="_qj0aQq-lEd-of-
gKNbzXRg" visible="false" type="6006">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_qj0aQ6-
lEd-of-gKNbzXRg" x="-98" y="34"/>
  </children>
  <styles xmi:type="notation:RoutingStyle" xmi:id="_qjqpQa-lEd-of-
gKNbzXRg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_qjqpQq-lEd-of-
gKNbzXRg" fontColor="16711680"/>
  <element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.5"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_qjqpQ6-
lEd-of-gKNbzXRg" points="[8, -2, 43, 8]${32, -7, 67, 3} ${32, -37, 67, -
27} ${8, -28, 43, -18}"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_N8W0EK-
mEd-of-gKNbzXRg" id="(0.9186046511627907,0.4050632911392405)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_qj0aRK-
lEd-of-gKNbzXRg" id="(0.7674418604651163,0.012658227848101266)"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_AlT7gK-mEd-of-gKNbzXRg"
type="4007" source="_KInlIK-jEd-of-gKNbzXRg" target="_L4hcgK-jEd-of-
gKNbzXRg">
```




```
<children xmi:type="notation:DecorationNode" xmi:id="_AldsgK-mEd-of-  
gKNbzXRg" type="6005">  
  <layoutConstraint xmi:type="notation:Location" xmi:id="_Aldsga-  
mEd-of-gKNbzXRg" x="-8" y="19"/>  
</children>  
  <children xmi:type="notation:DecorationNode" xmi:id="_Aldsgq-mEd-of-  
gKNbzXRg" visible="false" type="6006">  
    <layoutConstraint xmi:type="notation:Location" xmi:id="_Aldsg6-  
mEd-of-gKNbzXRg" y="60"/>  
  </children>  
  <styles xmi:type="notation:RoutingStyle" xmi:id="_ALT7ga-mEd-of-  
gKNbzXRg"/>  
  <styles xmi:type="notation:FontStyle" xmi:id="_ALT7gq-mEd-of-  
gKNbzXRg" fontColor="16711680"/>  
  <element xmi:type="IAC:Transition"  
href="IAC_sample.iac#//@transitions.6"/>  
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_ALT7g6-  
mEd-of-gKNbzXRg" points="[21, -2, -74, 0]${90, -2, -5, 0}"/>  
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_AldshK-  
mEd-of-gKNbzXRg" id="(0.7558139534883721,0.5189873417721519)"/>  
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_Aldsha-  
mEd-of-gKNbzXRg" id="(0.10638297872340426,0.4936708860759494)"/>  
</edges>  
  <edges xmi:type="notation:Edge" xmi:id="_cS2ZkK-mEd-of-gKNbzXRg"  
type="4007" source="_e7cT0K-jEd-of-gKNbzXRg" target="_foltsK-jEd-of-  
gKNbzXRg">  
    <children xmi:type="notation:DecorationNode" xmi:id="_cS2ZlK-mEd-of-  
gKNbzXRg" visible="false" type="6005">  
      <layoutConstraint xmi:type="notation:Location" xmi:id="_cS2Zla-  
mEd-of-gKNbzXRg" y="40"/>  
    </children>  
    <children xmi:type="notation:DecorationNode" xmi:id="_cS2Zlq-mEd-of-  
gKNbzXRg" visible="false" type="6006">  
      <layoutConstraint xmi:type="notation:Location" xmi:id="_cS2Zl6-  
mEd-of-gKNbzXRg" y="60"/>  
    </children>  
    <styles xmi:type="notation:RoutingStyle" xmi:id="_cS2Zka-mEd-of-  
gKNbzXRg"/>  
    <styles xmi:type="notation:FontStyle" xmi:id="_cS2Zkq-mEd-of-  
gKNbzXRg"/>  
    <element xmi:type="IAC:Transition"  
href="IAC_sample.iac#//@transitions.7"/>  
    <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_cS2Zk6-  
mEd-of-gKNbzXRg" points="[29, 2, -87, -7]${102, 8, -14, -1}"/>  
    <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_cS2ZmK-  
mEd-of-gKNbzXRg" id="(0.12962962962962962,0.4936708860759494)"/>  
</edges>  
  <edges xmi:type="notation:Edge" xmi:id="_d0mgMK-mEd-of-gKNbzXRg"  
type="4007" source="_foltsK-jEd-of-gKNbzXRg" target="_gTWh8K-jEd-of-  
gKNbzXRg">  
    <children xmi:type="notation:DecorationNode" xmi:id="_d0mgNK-mEd-of-  
gKNbzXRg" visible="false" type="6005">  
      <layoutConstraint xmi:type="notation:Location" xmi:id="_d0mgNa-  
mEd-of-gKNbzXRg" y="40"/>
```



```
</children>
<children xmi:type="notation:DecorationNode" xmi:id="_d0mgNq-mEd-of-
gKNbzXRg" visible="false" type="6006">
  <layoutConstraint xmi:type="notation:Location" xmi:id="_d0mgN6-
mEd-of-gKNbzXRg" y="60"/>
</children>
<styles xmi:type="notation:RoutingStyle" xmi:id="_d0mgMa-mEd-of-
gKNbzXRg"/>
<styles xmi:type="notation:FontStyle" xmi:id="_d0mgMq-mEd-of-
gKNbzXRg"/>
<element xmi:type="IAC:Transition"
href="IAC_sample.iac#/@transitions.8"/>
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_d0mgM6-
mEd-of-gKNbzXRg" points="[20, 0, -110, -6]${121, 5, -9, -1}"/>
<sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_d0wRMK-
mEd-of-gKNbzXRg" id="(0.8148148148148148,0.46835443037974683)"/>
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_d0wRMa-
mEd-of-gKNbzXRg" id="(0.10714285714285714,0.46835443037974683)"/>
</edges>
<edges xmi:type="notation:Edge" xmi:id="_fHTeMK-mEd-of-gKNbzXRg"
type="4007" source="_gTWh8K-jEd-of-gKNbzXRg" target="_g4wVEK-jEd-of-
gKNbzXRg">
  <children xmi:type="notation:DecorationNode" xmi:id="_fHTeNK-mEd-of-
gKNbzXRg" visible="false" type="6005">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_fHTeNa-
mEd-of-gKNbzXRg" y="40"/>
  </children>
  <children xmi:type="notation:DecorationNode" xmi:id="_fHTeNq-mEd-of-
gKNbzXRg" visible="false" type="6006">
    <layoutConstraint xmi:type="notation:Location" xmi:id="_fHTeN6-
mEd-of-gKNbzXRg" y="60"/>
  </children>
  <styles xmi:type="notation:RoutingStyle" xmi:id="_fHTeMa-mEd-of-
gKNbzXRg"/>
  <styles xmi:type="notation:FontStyle" xmi:id="_fHTeMq-mEd-of-
gKNbzXRg"/>
  <element xmi:type="IAC:Transition"
href="IAC_sample.iac#/@transitions.9"/>
  <bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_fHTeM6-
mEd-of-gKNbzXRg" points="[12, 1, -73, 0]${70, -1, -15, -2}"/>
  <sourceAnchor xmi:type="notation:IdentityAnchor" xmi:id="_fHdPMK-
mEd-of-gKNbzXRg" id="(0.8571428571428571,0.4177215189873418)"/>
  <targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_fHdPMa-
mEd-of-gKNbzXRg" id="(0.13513513513513514,0.45569620253164556)"/>
  </edges>
  <edges xmi:type="notation:Edge" xmi:id="_hf5GMK-mEd-of-gKNbzXRg"
type="4007" source="_g4wVEK-jEd-of-gKNbzXRg" target="_hiLskK-jEd-of-
gKNbzXRg">
    <children xmi:type="notation:DecorationNode" xmi:id="_hgC3MK-mEd-of-
gKNbzXRg" type="6005">
      <layoutConstraint xmi:type="notation:Location" xmi:id="_hgC3Ma-
mEd-of-gKNbzXRg" x="-6" y="-18"/>
    </children>
    <children xmi:type="notation:DecorationNode" xmi:id="_hgC3Mq-mEd-of-
```



```
gKNbzXRg" visible="false" type="6006">
  <layoutConstraint xmi:type="notation:Location" xmi:id="_hgC3M6-
mEd-of-gKNbzXRg" y="60"/>
</children>
<styles xmi:type="notation:RoutingStyle" xmi:id="_hf5GMa-mEd-of-
gKNbzXRg"/>
<styles xmi:type="notation:FontStyle" xmi:id="_hf5GMq-mEd-of-
gKNbzXRg" fontColor="16711680"/>
<element xmi:type="IAC:Transition"
href="IAC_sample.iac#//@transitions.10"/>
<bendpoints xmi:type="notation:RelativeBendpoints" xmi:id="_hf5GM6-
mEd-of-gKNbzXRg" points="[56, 1, -167, -3]$, [223, 4, 0, 0]"/>
<targetAnchor xmi:type="notation:IdentityAnchor" xmi:id="_hgC3NK-
mEd-of-gKNbzXRg" id="(0.0,0.5189873417721519)"/>
</edges>
</notation:Diagram>
```