

Designing Modeling Notations Readers Understand

THÈSE No 7261 (2016)

PRÉSENTÉE LE 15 septembre 2016

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE MODÉLISATION SYSTÉMIQUE
PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

George POPESCU

acceptée sur proposition du jury :

Prof. Bixio Rimoldi, président du jury
Prof. Alain Wegmann, directeur de thèse
Prof. Jeffrey Huang, rapporteur
Prof. Hendrik Knoche, rapporteur
Dr. Boris Fritscher, rapporteur



Suisse
2016

Acknowledgements

I would like to express my appreciation to the people who contributed to the success of this research.

First and foremost, I am grateful to my supervisor, Professor Alain Wegmann, whose inspiring ideas, extensive expertise and generous guidance taught me to see both the big picture and the details. Thank you, Alain, for supporting my work, broadening my research horizon, and challenging me! You continuously motivated me with your endless energy and joyful enthusiasm, and helped me develop my confidence in telling better stories with SEAM. I understood that what might seem evident deserves an analysis of deeper layers.

I am hugely indebted to my LAMS colleagues for making our lab a unique place to work in. Thank you, Dr. Gil Regev, for providing me with insightful feedback, helping me put my research into a larger perspective, sharing priceless consulting resources, and offering me the chance to review interesting papers that connect my work with other lines of research. Thank you, Ms. Gorica Tapandjjeva, for reviewing my work and proposing improvements countless number of times. Thank you, Ms. Aylin Jarrah Nezhad, for testing my models, and for pointing out implicit elements related to both the human and technology sides. Thank you, Ms. Aarthi Gopal, for your kind and creative feedback, and for making sure we all display a smile on our faces every single day. Thank you, Dr. Biljana Bajić-Bizumić, for our career discussions during which we explored consulting life. Thank you, Dr. Arash Golnam, for your challenging questions, and for guiding the synthesis of my write-up. Thank you, Dr. Anshuman Saxena, for checking my research progress, and, at every occasion, adding more pressure on to my shoulders. Thank you, Mr. Blaise Carrupt, for developing SeamCAD, and for allowing me to test numerous versions of my SEAM models. Thank you, Dr. Pierre Castori and Mr. Lucien Etzlinger, for developing TradeYourMind.com, that allowed me to understand SEAM modeling applied to business ideas. Thank you, Mrs. Danielle Alvarez and Mrs. Patricia Hejlt, for your administrative support. Thank you, Mrs. Holly Cogliati, for proofreading my English in this thesis. Thank you, Mrs. Angela Devenoge, for improving the German translation of my thesis abstract. I am grateful to all the students who helped me to understand my thesis better. Thank you all for making LAMS an incredible place for learning and working. The Ph.D. research years we spent together were some of the best of my life.

I would like to acknowledge the members of my Ph.D. jury. I am thankful to Professor Bixio Rimoldi for having kindly accepted the role as the committee president. I am grateful to Professor Hendrik Knoche for directing my attention to the relevant pieces of literature related to user-

centered design, and for helping me to formulate better significant research results. I also address my thanks to Professor Jeffrey Huang and Professor Boris Fritscher who gave me important feedback on my work.

I cannot thank enough the people who inspired and helped me through my professional life. They helped me steer my career journey harmoniously, in the direction of my competencies. Thank you, Professor Radu Gologan, for being my role-model, inspiring me to connect the mathematical and real worlds together. Thank you, Professor Simona Caramihai, for your trust in my engineering skills for my bachelor's thesis. Thank you, Professor Daniel Krob, for considering me as one of the finest students that you worked with for my master's program. Thank you, Mr. Preben Lund, for agreeing with my proposal to do a Ph.D. at EPFL, one of the most prestigious academic institutions in the world. Thank you, Mr. Claudiu Balan, my dear friend and colleague, for your patience and precious contribution to our first research papers together.

My friends and colleagues from EPFL made my integration in Switzerland such a pleasant experience. I'd like to especially thank Dr. Claudiu Muşat, Mr. Bogdan Stoica, Dr. Florin Dinu, Mr. Jaufray Sornette, Dr. Mihai Dobrescu, Dr. Dan Alistarh, Ms. Sonia Bogoş, Mr. Sergiu Găman, Dr. Flaviu Roman, Dr. Alexandru Rusu, Dr. Nicolae Chiurtu, Dr. Tudor Calistru, Mr. Haprian Vlad, Dr. Francisco Macedo, Mr. Elio Abi Karam, Ms. Rafah El-Khatib, Ms. Lavinia Ghiţa, Ms. Corina Stoenescu, Ms. Raluca Constantin, and Ms. Zerrin Celebi for their cooperation and positive energy that were essential for the completion of the field work. It was a pleasure to develop exploratory discussions about the advancements of technology in our society with Dr. Jean-Yves Henry.

My close friends from Geneva, Ms. Diana Mihai, Mrs. Ligia Folea and Mr. Thomas Barzan, Mrs. Raluca Paşol and Mr. Bogdan Paşol, and Mrs. Ruxandra Năftănăilă-Postolache and Mr. Eugen Naftănăilă, I would like to thank you for all the excitement and experiences we shared together at our cozy homes, out in the wild nature, or on the icy slopes.

My tennis team at Saint Sulpice greatly contributed to my well-being, relaxation, and my ambition of winning by overcoming obstacles. I am grateful to Mr. Frédéric Bonvin, Mr. André Gruhl, Mr. Maxime Schlaepfer, Mr. Jonas Burdel, Mr. Matthieu Humair, Mr. Matyas Lindenberger, Mr. André Monnier, Mr. Marc-Antoine Habisreutiger, and other team members for the evening practice, delightful apéros, and good mood after our InterClub wins.

I owe great appreciation from the bottom of my heart to my entire family. Thank you, Ms. Ginelă Popescu, my sister, for explaining how an excellent doctor helps humans every day, to infer what a doctor in computer science working with computers should do to serve humanity as well. Thank you, Mrs. Ana Popescu and Mr. Paul Popescu, my parents, for teaching me the right values in life, for offering me the next generation of computers every 3-4 years, and for encouraging me to explore my interests based on a solid educational background. Thank you, Mrs. Elena Popescu and Mr. Ilie Popescu, my grandparents, for taking excellent care of my first 7 years of life, which were full of freedom and passion for exploring the world. Thank you, Mrs. Simona Neagu and Cristian Neagu, my parents-in-law, for loving me unconditionally, as your own son. Thank you, Mrs. Victoria Ene, my grand-mother in law, for your endless care and positive attitude.

My greatest thanks I address to my wife, Mrs. Monica-Maria Popescu-Neagu, who has been by my side during both successful and challenging moments. Thank you for your loving encouragements, patience, understanding, and corporate-inspired advice. You contributed to my professional and personal achievements since the day we met.

Lausanne, the 30th of September 2016

The measure of greatness in a scientific idea
is the extent to which it stimulates thought and opens up new lines of research.

— Paul A.M. Dirac

Humility is the awareness that there's a lot you don't know.
And that a lot of what you think you know, is distorted, or wrong.

This is the way humility leads to wisdom.

— David Brooks

Abstract

[Context] IT modelers create models to communicate their conceptualization of an organization and to facilitate the collaboration between business and IT people. The story they convey in their models is how the corporate IT systems fulfill the business people's needs. The business people who must validate these models often have no IT background. Therefore, IT modelers need to create models that non-IT individuals understand.

[Motivation and Problem] We explore the understanding of models created with IT notations by non-IT individuals. We enable IT modelers to create models such that non-IT readers can understand. We also enable IT notation designers to design notations that help modelers to create models that non-IT readers can understand.

[Idea and Results] We perform an explorative qualitative case-study for two modeling notations, SEAM and i*, by using two cases: car-maintenance service and meeting scheduler, in order to investigate how readers interpret models. Readers evaluate our models and identify the elements that are difficult to comprehend. We create improved models by making these elements easier to comprehend, so that readers understand the modeler's story. We reduce the misalignment between the modeler's story and the readers' perceptions of this story. We improve models without changing the identity of the underlying notation.

[Contributions] Our contributions address modelers and designers of IT notations.

For IT modelers, such as IT architects, service designers, and consultants, using IT notations (e.g., UML, BPMN, ArchiMate), this research provides the means necessary to create models so that readers, unfamiliar with the notation, understand. We recommend to modelers to model: (i) the relation with reality, by focusing on the readers' conceptualizations, (ii) the rationale, by using the questions, options, criteria and their assessment, and (iii) the story, by using story-phases: context, conflict, climax and closure. In addition, we provide them with a modeling process for creating models.

For designers of IT notations, this research provides the means necessary to design notations. We recommend to designers to design: (i) the relation with reality, by identifying implicit elements in the modeling notation by evaluating readers' understanding of what elements represent, (ii) the rationale, by using visual cues to guide readers' understanding of the problem, and (iii) the story, by

creating models in which different stories are told to elicit readers' understanding of what happens in the model. In addition, we provide them with a design process for designing notations.

Keywords

Case study, Model, Story, Modeler, Reader, Designer, Notation

Résumé

[Contexte] Les modélisateurs IT créent des modèles pour communiquer leur conceptualisation d'une organisation et pour faciliter la collaboration entre les personnes de formation métier et IT. L'histoire qu'ils montrent dans leurs modèles est comment les systèmes informatiques d'entreprise répondent aux besoins des personnes métiers. Les personnes métiers qui doivent valider ces modèles n'ont souvent pas une formation IT. Par conséquent, les modélisateurs IT ont besoin de créer des modèles que les personnes non-IT comprennent.

[Motivation et Problème] Nous explorons la compréhension des modèles créés avec des notations IT par des personnes non-IT. Nous aidons les modélisateurs IT de créer des modèles tels que les lecteurs non-IT peuvent comprendre. Nous aidons également les concepteurs des notations IT de concevoir des notations qui aident les modélisateurs à créer des modèles que les lecteurs non-IT peuvent comprendre.

[Idée et Résultats] Nous effectuons une étude de cas qualitative et exploratoire pour deux notations de modélisation, SEAM et i*, en utilisant deux cas : un service d'entretien d'une voiture et un planificateur de réunions, afin d'examiner comment les lecteurs interprètent les modèles. Les lecteurs évaluent nos modèles et identifient les éléments qui sont difficiles à comprendre. Nous créons des modèles améliorés en rendant ces éléments de manière plus facile à comprendre, de sorte que les lecteurs comprennent l'histoire du modélisateur. Nous réduisons le désalignement entre l'histoire du modélisateur et les perceptions des lecteurs de cette histoire. Nous améliorons les modèles sans changer l'identité de la notation sous-jacente.

[Contributions] Nos contributions sont adressées aux modélisateurs et aux concepteurs de notations IT.

Pour les modélisateurs IT, tels que les architectes IT, les concepteurs de services, et les consultants, qui utilisent des notations IT (par exemple, UML, BPMN, ArchiMate), cette recherche fournit les moyens nécessaires pour créer des modèles de sorte que les lecteurs, peu familiers avec la notation, comprennent. Nous recommandons aux modélisateurs de modéliser : (i) la relation avec la réalité, en mettant l'accent sur les conceptualisations des lecteurs, (ii) la justification de la conception, en utilisant les questions, les options, les critères et leurs évaluations, et (iii) l'histoire, en utilisant les phases d'une histoire : le contexte, le conflit, le point culminant, et la situation finale. En outre, nous les proposons un processus de modélisation pour la création de modèles.

Pour les concepteurs de notations IT cette recherche fournit les moyens nécessaires pour concevoir des notations. Nous recommandons aux concepteurs de concevoir : (i) la relation avec la réalité, en identifiant les éléments implicites dans la notation de modélisation en évaluant la compréhension des lecteurs sur ce que les éléments représentent, (ii) la justification de la conception, en utilisant des symboles graphiques pour guider la compréhension des lecteurs concernant le problème, et (iii) l’histoire, en créant des modèles dans lesquels différentes histoires sont racontées pour observer la compréhension des lecteurs de ce qui se passe dans le modèle. En outre, nous leur proposons un processus de design pour le design de notations.

Mots-clés

Etude de cas, Modèle, Histoire, Modélisateur, Lecteur, Concepteur, Notation

Zusammenfassung

[Kontext] IT Modellierer schaffen Modelle, um ihre Konzeptualisierung einer Organisation zu veröffentlichen und um die Zusammenarbeit zwischen Geschäfts- und IT-Mitarbeitern zu fördern. Das Schema, das sie in den Modellen vortragen, zeigt wie die Organisations-IT-Systeme die Bedürfnisse der Unternehmer erfüllen. Die Unternehmer, die diese Modelle validieren müssen, haben oft keine IT-Bildung. Deshalb müssen IT-Modellierer Modelle erstellen, die Unternehmer ohne IT-Bildung verstehen.

[Motivation und Problem] Wir erforschen, wie Geschäftsleute (ohne IT-Bildung) Modelle, die mit IT-Notationen erstellt wurden, verstehen. Wir ermöglichen IT-Modellierern Modelle zu erstellen, die Geschäftsleute verstehen können. Wir ermöglichen auch Designern von IT-Notationen zu entwerfen, die Modellierern helfen, Modelle zu erstellen, die Geschäftsleute verstehen können.

[Idee und Ergebnisse] Wir führen eine explorative qualitative Fallstudie für zwei Modellierungsnotationen, SEAM und i^* , anhand zweier Beispiele durch: Auto-Wartungs-Service und Terminplaner, um zu untersuchen, wie die Leser Modelle interpretieren. Die Leser bewerten unsere Modelle und bestimmen die Elemente, die schwer zu verstehen sind. Wir schaffen bessere Modelle, indem wir diese Elemente leichter verständlich machen, so dass die Leser das Schema der Modellierer verstehen. Wir verringern die Diskrepanz zwischen dem Schema der Modellierer und deren Wahrnehmung durch die Leser. Wir verbessern Modelle, ohne die Identität der zugrundeliegenden Notation zu ändern.

[Beitrag] Unsere Beiträge sind für Modellierer und Designer von IT-Notationen bestimmt.

Für IT-Modellierer, wie IT-Architekten, Service-Designer und Berater, die IT-Notationen nutzen (z. B. UML, BPMN, ArchiMate), liefert diese wissenschaftliche Arbeit die notwendigen Mittel um Modelle zu schaffen, die für Leser, die nicht mit der Notation vertraut sind, verständlich sind. Wir empfehlen Modellierern folgendes zu modellieren: (i) den Bezug zur Realität, mit den Konzeptualisierungen der Leser, (ii) die Begründung, mit den Fragen, Optionen, Kriterien, und deren Bewertung, und (iii) das Schema, mit den Inhalten: Kontext, Konflikt, Höhepunkt, und Abschluss. Außerdem bieten wir ihnen einen Modellierungsprozess um Modelle zu schaffen.

Für Designer von IT-Notationen, liefert diese wissenschaftliche Arbeit die notwendigen Mittel um Notationen zu entwerfen. Wir empfehlen Designern folgendes zu entwerfen: (i) den Bezug zur Realität, durch die Identifikation der impliziten Elemente in der Modellierungsnotation, basierend auf

der Bewertung der Leser, über das Verständnis, was die Elemente darstellen, (ii) die Begründung, durch die Verwendung von visuellen Hinweisen um das Verständnis der Leser vom Problem zu unterstützen, und (iii) das Schema, durch die Schaffung von Modellen, in denen verschiedene Schemen erzählt werden, die dem Leser verständlich machen, was die Modelle aussagen wollen. Außerdem bieten wir ihnen einen Designprozess um Notationen zu entwerfen.

Stichworte

Fallstudie, Modell, Schema, Modellierer, Leser, Designer, Notation

Contents

List of Figures	xiii
List of Tables	xvii
Chapter 1 Introduction	1
1.1 Main Concepts	3
1.2 Motivation.....	4
1.3 Architecture and IT Architecture.....	4
1.4 Thesis Organization	5
Chapter 2 The State of the Art	7
2.1 IT Modeling Notations	7
2.1.1 Systemic Enterprise Architecture Method	8
2.1.2 i*.....	19
2.1.3 Unified Modeling Language	22
2.1.4 Concluding Remarks.....	25
2.2 Evaluation of Modeling Notations.....	26
2.2.1 Semiotic Quality Framework	26
2.2.2 Seven Process Modeling Guidelines	27
2.2.3 Cognitive Dimensions of Notations	27
2.2.4 Physics of Notations.....	28
2.2.5 Concluding Remarks.....	30
2.3 Model Design Frameworks.....	31
2.3.1 Design Analysis	31
2.3.2 Story Phases	32
2.3.3 Concluding Remarks.....	33
2.4 Summary.....	33
Chapter 3 Research Method	35
3.1 Model Creation and Notation Design	35
3.2 Research Relevance	38
3.3 Research Rigor.....	40
3.4 Research Process	41
3.5 Interviews	44

3.6	Summary.....	45
Chapter 4	Application to SEAM	47
4.1	Research Protocol.....	47
4.2	Research Design.....	47
4.3	Interviews.....	48
4.4	Feedback.....	48
4.4.1	Limited Relation with Reality.....	49
4.4.2	No Rationale and No Choice.....	50
4.4.3	No Story.....	50
4.5	Contributions to SEAM modelers.....	51
4.5.1	Relation with Reality.....	51
4.5.2	Rationale and Choice.....	57
4.5.3	Story.....	61
4.6	Summary.....	69
Chapter 5	Application to i*	71
5.1	Research Protocol.....	71
5.2	Research Design.....	71
5.3	Interviews.....	72
5.4	Feedback.....	73
5.4.1	Limited Relation with Reality.....	73
5.4.2	Partial Rationale.....	73
5.4.3	No Story.....	73
5.5	Contributions to i* modelers.....	74
5.5.1	Relation with Reality.....	74
5.5.2	Rationale.....	78
5.5.3	Story.....	80
5.6	Summary.....	89
Chapter 6	Theoretical Principles and Contributions	91
6.1	The Need for Modeling the Relation with Reality.....	91
6.1.1	The Identification of the Zone of Proximal Development.....	92
6.1.2	The Alignment of Interpretations.....	96
6.1.3	The Agreement on Boundaries.....	98
6.2	The Need for Modeling the Rationale.....	99
6.2.1	The Search for Satisficing Options.....	101
6.2.2	The Accommodation of Needs.....	101
6.3	The Need for Modeling the Story.....	102
6.3.1	The Problematic Situation Modeling Process.....	104
6.3.2	The Comics Modeling Process.....	105

6.3.3	The Story Design Process	105
6.3.4	The Animation Design Process.....	106
6.3.5	The Narrative Storyboards Design Process	106
6.3.6	The Comparison of SEAM and i* Storyboards	107
6.4	Contributions to Modelers	108
6.4.1	Recommendations	108
6.4.2	Modeling Process	109
6.4.3	Observations for SEAM Modelers.....	110
6.4.4	Concluding Remarks.....	111
6.5	Contributions to Designers	112
6.5.1	Recommendations	112
6.5.2	Design Process	113
6.5.3	Observations for SEAM Designers.....	113
6.5.4	Concluding Remarks.....	115
6.6	Discussions of Recommendations with SEAM Modelers.....	116
6.7	Limitations.....	117
6.8	Summary.....	118
Chapter 7	Conclusions and Future Work	120
7.1	Achieved Results	120
7.2	Future Developments.....	121
Glossary		123
Bibliography		125
Appendix 1 Case of Car-Maintenance Service with SEAM.....		138
Appendix 2 Case of Meeting Scheduler with i*.....		149
Appendix 3 Case of Business Strategy at SITRA with SEAM		153
Appendix 4 Case of IT Strategy at EPFL with SEAM.....		161
Appendix 5 Current and proposed SEAM Evaluation with the Physics of Notations		169
Appendix 6 Other IT Modeling Notations.....		172
Object-Process Methodology and Notation		172
Design and Engineering Methodology for Organizations.....		174
Soft Systems Methodology and Rich Pictures		176
Business Process Modeling Notation.....		179
ArchiMate.....		180
Business Motivation Model		182
Use Case Map.....		183
e ³ -value		184
Appendix 7 Preliminary Study on Service Modeling		185
Curriculum Vitae.....		190

List of Figures

Figure 1: SEAM Behavior model showing a hierarchy of systems and their behavior.....	10
Figure 2: SEAM Goal-Belief model example	11
Figure 3: SEAM-Behavior model example	12
Figure 4: SEAM-Supplier-Adopter-Relationship model and matrix example	13
Figure 5: SEAM modeler’s conceptualization.....	14
Figure 6: SEAM model of SITRA using the workshop notation.....	16
Figure 7: SEAM model of SITRA using the academic notation	17
Figure 8: SEAM model in TradeYourMind done with the online notation – source: (Etzlinger, Castori, & Wegmann, 2016).....	18
Figure 9: SEAM model of TradeYourMind using the academic notation.....	18
Figure 10: i* notation – source: (Yu E. , University of Toronto, 2001)	20
Figure 11: i* Strategic Dependency (SD) model example – source: (Yu E. S., 1997).....	21
Figure 12: i* Strategic Rationale (SR) model example – source: (Yu E. S., 1997).....	21
Figure 13: Unified Modeling Language model types – source: (Unified Modeling Language, 2016)	23
Figure 14: Unified Modeling Language activity diagram notation and model example – source: (Fowler, 2004)	25
Figure 15: Semiotic Quality framework – source: (Krogstie J. S., 2003)	26
Figure 16: Physics of Notations principles applied to a vehicle example	30
Figure 17: Question, Options, Criteria framework – adapted from (MacLean, Young, Bellotti, & Moran, 1991)	32
Figure 18: Information-Systems research-framework with the addition of the practical and research questions – source: (Hevner, March, Park, & Ram, 2004)	36
Figure 19: Information-Systems research-framework applied to our research with the addition of the two sets of practical and research questions	36
Figure 20: Research process	41
Figure 21: Service models drawn by people using an ad-hoc notation	42
Figure 22: Interview with one participant interpreting a model	45
Figure 23: SEAM initial model.....	48
Figure 24: Modeler's goal and model information with the proposed SEAM	52
Figure 25: Modeler’s viewpoints and story summary from model iterations.....	52
Figure 26: Human, object and organization actors with the current SEAM.....	52
Figure 27: Human, object and organization actors with the proposed SEAM	52
Figure 28: Human, object and organization actors from model iterations.....	53
Figure 29: Graphical metaphors with the proposed SEAM.....	54
Figure 30: Graphical metaphors from model iterations	54
Figure 31: Service offering with the current SEAM.....	55
Figure 32: Service offering with the proposed SEAM	55

Figure 33: Service offering from model iterations	56
Figure 34: Terminology with the current SEAM.....	57
Figure 35: Terminology with the proposed SEAM	57
Figure 36: Terminology from model iterations with SEAM	57
Figure 37: Expression of rationale from model iterations	59
Figure 38: Questions, Options and Criteria framework extension in SEAM	60
Figure 39: Choice rationale based on Questions, Options and Criteria with the proposed SEAM	60
Figure 40: Storyboard design for each SEAM model instance based on the four story-phases	61
Figure 41: The first model instance of the proposed SEAM model (context).....	62
Figure 42: The second model instance of the proposed SEAM model (conflict).....	63
Figure 43: The third model instance of the proposed SEAM model (conflict)	64
Figure 44: The fourth model instance of the proposed SEAM model (conflict).....	65
Figure 45: The fifth model instance of the proposed SEAM model (climax)	66
Figure 46: The sixth model instance of the proposed SEAM model (climax)	67
Figure 47: The seventh model instance of the proposed SEAM model (closure)	68
Figure 48: Contributions to SEAM - from the current SEAM to the proposed SEAM.....	70
Figure 49: i^* initial model – source: (Yu E. S., 1997).....	72
Figure 50: Modeler’s goal and model information with the proposed i^*	74
Figure 51: Human and IT actors with the current i^*	75
Figure 52: Human and IT actors with the proposed i^*	75
Figure 53: Human and IT actors from model iterations	75
Figure 54: Goals, tasks, sub-tasks, softgoals and resources with the current i^*	76
Figure 55: Goals, tasks, sub-tasks, softgoals and resources with the proposed i^*	76
Figure 56: Goals, tasks, softgoals and resources from model iterations.....	76
Figure 57: Terminology with the current i^*	77
Figure 58: Terminology with the proposed i^*	77
Figure 59: Terminology from model iterations with i^*	78
Figure 60: Questions, Options and Criteria framework extension in SEAM	79
Figure 61: Rationale based on Question, Option and Criteria with the proposed i^*	80
Figure 62: Storyboard design for each i^* model instance based on the four story-phases	80
Figure 63: The first model instance of the proposed i^* model (context).....	82
Figure 64: The second model instance of the proposed i^* model (conflict).....	83
Figure 65: The third model instance of the proposed i^* model (conflict)	84
Figure 66: The fourth model instance of the proposed i^* model (conflict).....	85
Figure 67: The fifth model instance of the proposed i^* model (climax)	86
Figure 68: The sixth model instance of the proposed i^* model (climax)	87
Figure 69: The seventh model instance of the proposed i^* model (closure)	88
Figure 70: Contributions to i^* - from the current i^* to the proposed i^*	90

Figure 71: Relations between reality, conceptualization and model – adapted from (Regev, 2003) ...	92
Figure 72: Zone of proximal development between the reader and the modeler	93
Figure 73: Relations between reality, conceptualization and one model for SEAM.....	95
Figure 74: The Design Rationale between the modeler and the readers.....	100
Figure 75: 6-step process for story design – source: (McCloud, 1994).....	105
Figure 76: Model that is in accordance with the identity of the SEAM notation	114
Figure 77: Model that is not in accordance with the identity of the SEAM notation	114
Figure 78: SEAM car-maintenance service model showing the car dealer as whole and as composite	138
Figure 79: SEAM car-maintenance service model showing the two dealers as wholes.....	139
Figure 80: SEAM car-maintenance service model showing one dealer as a composite.....	139
Figure 81: SEAM car-maintenance service model showing wholes and composites more explicitly	140
Figure 82: SEAM car-maintenance service model showing the composition of one dealer	141
Figure 83: SEAM car-maintenance service model showing the customers to the left and the dealers to the right.....	141
Figure 84: SEAM car-maintenance service model showing the car with the hood open at the dealer during service	141
Figure 85: SEAM car-maintenance service model showing the implementation of the service by the dealer	142
Figure 86: SEAM car-maintenance service model showing the exchange of money for the serviced car	142
Figure 87: SEAM car-maintenance service model showing back the customers to the right and the suppliers to the left	143
Figure 88: SEAM car-maintenance service model showing the choice confirmation for one dealer	143
Figure 89: SEAM car-maintenance service model showing the low-cost implementation justified by the infrastructure	144
Figure 90: SEAM car-maintenance service model showing the completion of the service with payment	144
Figure 91: SEAM car-maintenance service model showing the goals and beliefs of the two customers separately	145
Figure 92: SEAM car-maintenance service model showing the distinction between the dealer shown as whole (photo from the outside) and as composite (photo from the inside).....	145
Figure 93: SEAM car-maintenance service model showing the choice of dealers with two processes	146
Figure 94: SEAM car-maintenance service model showing viewpoints and visual metaphors	147
Figure 95: SEAM car-maintenance service model showing the Questions, Options and Criteria mapping	148
Figure 96: i* meeting scheduler model using color coding	149
Figure 97: SEAM meeting scheduler model.....	150
Figure 98: i* meeting scheduler model with the current i* notation using structuring – adapted from (Yu E. S., 1997).....	151
Figure 99: i* meeting scheduler model with add-ins for the rationale – adapted from (Yu E. S., 1997)	152
Figure 100: SEAM model of SITRA using the academic notation	153
Figure 101: SEAM model of SITRA using a notation based on photos and logos	154
Figure 102: SEAM model of SITRA combining Behaviour and Goal-Belief modeling	155
Figure 103: SEAM model of SITRA using the artwork of a professional designer.....	156

Figure 104: SEAM model of business strategy at SITRA – Context, conflict and climax – source: (Wegmann A. , 2012)	157
Figure 105: SEAM service organization template sketch.....	161
Figure 106: Storyboard of SEAM model instances showing the IT strategy at EPFL	162
Figure 107: SEAM model of IT strategy at EPFL - Context – source: (Wegmann & Rey, 2016).....	163
Figure 108: SEAM model of IT strategy at EPFL - Conflict – source: (Wegmann & Rey, 2016)	164
Figure 109: SEAM model of IT Strategy at EPFL - Climax A – source: (Wegmann & Rey, 2016) .	165
Figure 110: SEAM model of IT Strategy at EPFL - Climax B – source: (Wegmann & Rey, 2016) .	165
Figure 111: SEAM model of IT Strategy at EPFL - Closure – source: (Wegmann & Rey, 2016)	166
Figure 112: SEAM model of IT Strategy at EPFL – Problem.....	167
Figure 113: SEAM model of IT Strategy at EPFL – First step of the solution	168
Figure 114: SEAM model of IT Strategy at EPFL – Second step of the solution.....	168
Figure 115: Object-Process Methodology notation – source: (Dori, 2002).....	172
Figure 116: Object-Process Methodology model example – source: (Dori, 2002)	173
Figure 117: Design and Engineering Methodology for Organizations model example – source: (van Reijswoud, Mulder, & Dietz, 2001).....	175
Figure 118: Hard and Soft Systems Methodology – source: (Checkland, 1990)	176
Figure 119: Soft Systems Methodology model example using the Rich Picture notation – source: (Monk & Howard, 1998)	177
Figure 120: Business Process Modeling Notation notation– source: (ConceptDraw, 2016)	179
Figure 121: Business Process Modeling Notation model example - source: (ConceptDraw, 2016)..	179
Figure 122: ArchiMate notation – source: (The Open Group, 2016)	180
Figure 123: ArchiMate model example – source: (Kraan, 2016)	181
Figure 124: Business Motivation Model notation	182
Figure 125: Business Motivation Model model example	182
Figure 126: Use Case Map notation.....	183
Figure 127: Use Case Map model example	183
Figure 128: e ³ -value notation	184
Figure 129: e ³ -value model example.....	184
Figure 130: Demographics analysis of the preliminary study	185
Figure 131: Average number of entities and connections between entities in participants’ services models	187
Figure 132: Concept analysis of participants’ service models.....	187
Figure 133: Evaluation of participants' service models with the Physics of Notations principles	189

List of Tables

Table 1: Graphical symbols used for the SEAM Goal-Belief model	11
Table 2: Graphical symbols used for the SEAM Behavior model.....	12
Table 3: Graphical symbols used for the SEAM Supplier-Adopter-Relationship model.....	13
Table 4: Seven Process Modeling Guidelines – source: (Mendling, Reijers, & van der Aalst, 2009) .	27
Table 5: Cognitive Dimensions of Notations – source: (Green, Blandford, Church, Roast, & Clarke, 2006)	28
Table 6: Physics of Notations – source: (Moody D. L., 2010)	28
Table 7: Information-Systems research-framework applied to the current research – source: (Hevner, March, Park, & Ram, 2004)	38
Table 8: Case study applied to two modeling notations: SEAM and i*	46
Table 9: Question, Options and Criteria for the two customers	58
Table 10: Question, Option and Criteria for the meeting initiator and participants	78
Table 11: Modeling notations and corresponding topics	121
Table 12: Current SEAM notation evaluation with PoN	169
Table 13: Proposed SEAM notation evaluation with PoN	170
Table 14: Summary of main differences between business and IT participants' models.....	188

Chapter 1 Introduction

IT modelers create graphical models to design IT systems. These models illustrate the services that the enterprise IT systems provide to their stakeholders. Models are used to show and develop the collaboration between people, and are shared with stakeholders. The stakeholders need to understand them in order to ensure that their expectations and needs are met. However, stakeholders often do not have an IT background and, therefore, have difficulties understanding the models.

For example, IT architects often organize workshops with business stakeholders. During these workshops, the architect creates models to discuss organizational strategy with stakeholders. These models include entities related to IT systems from the architect's and stakeholders' environments and are used to show and develop the collaboration between participants. The workshop participants (as model readers) often like to listen to a concrete story. They believe that they understand the message (or story) that the IT architect (modeler) conveyed in the model. Some participants (with an IT background), familiar with the model's underlying modeling notation, can correctly interpret the modeler's story. Other participants (with a non-IT background), not familiar with the notation, frequently encounter difficulties interpreting this story. They perceive a different story than the story that the modeler wanted to convey. Also, after the workshop, both types of readers are often unable to understand the story that the modeler conveyed during the workshop. They do not remember the IT modeler's explanations and the story becomes difficult to understand (implicit). Models can be done with pen and paper during workshops, but they are too vague and cryptic for readers. Or they can be done with modeling tools after the workshop, but they are too complicated for readers.

IT modelers use different IT notations (e.g., SEAM, i*, UML, BPMN, ArchiMate) to create models for non-IT people. In these models they show how people work together with the IT (organizational aspects), rather than the IT itself. As experts, they assume a common understanding of their notation by all model readers. We formulate the following problem: How should the modeler (e.g., IT architect) create models so that readers (e.g., non-IT workshop participants) understand the modeler's story? We are interested in enabling IT modelers to create models such that non-IT readers can understand the story, as intended by the modeler, only by looking at the model (without the modeler's audio / video intervention). Such models may be sent by the modeler after the workshop for the readers.

The challenge is to improve the model's underlying notation while maintaining its identity, that is expressed with a set of modeling principles applied to the notation's graphical symbols that allow modelers to create models in a certain way. These principles render the notation recognizable to modelers and readers. Notation designers want to promote their notation to modelers and readers in order to recognize this notation in the models that modelers create with or for readers. Therefore, we do not design new notations, but rather propose improved notations that are more understandable for readers and have the same identity as the original ones.

In this thesis, we assume that the modeler wants to convey a story using a graphical model, for instance communicate business strategy in a workshop. Graphical models are relatively easy to create compared to video, and more interesting for readers to explore the modeler's story than text or audio. We argue that a model is richer than one thousand words, enabling the readers to understand the modeler's story. Every reader can perceive a story by observing a model, but this story might only partly correspond to the one that the modeler wants to tell. Consequently, the main idea of this work is to enable modelers to create understandable models. We create these models by learning how readers interpret models.

We study two IT modeling notations, SEAM and i^* , by modeling two non-IT cases that non-IT people can easily relate to. The insights we gained based on readers' interpretations of our models are grouped in three categories: relation with reality ("what do the elements represent?"), rationale ("what is the problem?") and story ("what happens?").

We propose model improvements in a series of model iterations in which we make more explicit the relation with reality, the rationale and the story. IT modelers need to create models that are based on commonly encountered situations, not only models that describe the in-depth functionality of IT systems. If they would create models of IT systems, then non-IT readers would not understand these models.

This work provides the means necessary for modeling experts (IT modelers), such as IT architects, service designers, consultants, and business and IT analysts to create models for non-experts (non-IT readers), while maintaining the identity of the model's underlying IT notation. The contribution of this work is the design of improved notations useful to create improved models to communicate better with readers.

We propose recommendations to IT modelers. These recommendations concentrate on modeling (i) the relation with reality, by focusing on the readers' conceptualizations of their observed realities instead of the modeler's observed reality, (ii) the rationale, by using the questions, options, criteria and their assessment expressed explicitly in the model, and (iii) the story, by using the context, conflict, climax and closure story-phases. In addition, we provide modelers with a modeling process that they can use to create improved models.

We propose recommendations to designers of IT notations. These recommendations concentrate on designing (i) the relation with reality, by identifying implicit elements in the modeling notation by evaluating readers' understanding of what elements represent, (ii) the rationale, by using visual cues to guide readers' understanding of the problem, and (iii) the story, by creating models in which different stories are told to elicit readers' understanding of what happens in the model. In addition, we provide designers with a design process that they can use to design improved notations.

In the next section, we clarify the main concepts used throughout the thesis, such as model, story, modeler, reader, designer and notation. Then, we explain our motivation for this work. We present an analogy between architecture and IT architecture and conclude with a brief summary of the organization of the thesis.

1.1 Main Concepts

The main concepts used in this research are defined below. Besides these most frequently used concepts, other related terms are defined in the glossary, at the end of this work.

The author of this research has multiple roles: interviewer, SEAM modeler, i* modeler, actor in a SEAM model, and designer of modeling notations.

By **interviewer**, we mean the person who carries out interviews, the research investigator. He gathers feedback from interviewees (readers) in order to evaluate how they understand the modeler's story.

By **modeler**, we mean the person who creates a model, the author of the model.

By **IT modeler**, we mean the modeler who uses an IT notation to create models.

By **model**, we mean a graphical representation, based on a notation, of a conceptualization of reality of a modeler that is used to show a story to readers. It is a schematic description of something, a system or phenomenon that accounts for its properties, and is used to study its characteristics and show its construction (appearance) and behavior (action). A model is used to show a story of a modeler. A model instance reflects a section of this story. A model element is shown using a graphical symbol in one or more model instance(s).

By **improvement**, we mean a change of the model to make it more understandable for readers, i.e. so that readers interpret the story that the modeler wants to show.

By **story**, we mean a sequence of events shown in the model that is based on the modeler's conceptualization of the observed reality and that is interpreted by readers; what readers perceive in the model. Readers might interpret a different story than the story that the modeler wants to convey.

By **reader**, we mean the person, unfamiliar with the modeling notation, who interprets a model through his own perceptions.

By **notation**, we mean a set of graphical symbols (visual vocabulary) and composition rules (visual grammar) used by modelers to create a model. The notation is based on a method.

By **IT notation**, we mean a notation used in IT, such as SEAM, i*, UML, BPMN, ArchiMate.

By **designer**, we mean the person who designs a modeling notation. He analyzes how modelers create their models using a notation. He helps modelers create models that readers can understand.

By **IT designer**, we mean the designer who designs IT notations.

Modelers use **modeling notations** to create **models** that comprise **model instances** formed with **graphical elements** to show **stories** to **readers**. **Designers** design **improved notations** that help **modelers** create **improved models** by using these **notations**.

1.2 Motivation

We are interested to understand how non-IT readers (readers unfamiliar with the IT notation) interpret models created by IT modelers (experts with the IT notation). We consider two IT modeling notations, SEAM and i^* , to create models based on everyday examples. This is inspired by the fact that IT modelers mostly need to communicate to non-IT people how people work together with the IT rather than describe the IT itself, for instance discuss business strategy in the context of a workshop. If IT modelers would create IT models, non-IT readers would not understand these models. For IT readers, IT modelers may create IT models. We investigate, “How should a SEAM / i^* modeler create a (SEAM / i^*) model so that (non-IT) readers understand the modeler’s story?”. We choose SEAM because the thesis author was educated on its use and because he was asked by a SEAM modeler to improve the SEAM notation. We choose i^* because it is a widely used notation in requirements engineering that is applied to early requirements, similarly to SEAM. Other IT notations (e.g., UML, BPMN, ArchiMate) are presented in Appendix 6. Our initial presentation of these notations serves as a starting point in applying the research process included in this thesis to improve them.

We want to model as explicitly as possible the elements that guide readers understanding of the story that the modeler wants to show. We aim to do so by evaluating and designing models iteratively with readers. First, we want to interview readers to test if they understand our initial model. Second, we want to collect readers’ suggestions on how to improve the model. Third, we want to implement readers’ suggestions in a new model iteration. The suggestions that are considered are the ones that maintain the identity (set of principles) of the model’s underlying notation because we want to propose improvements to notations so that modelers recognize them rather than create new notations. In repeating these steps, we identify elements that are difficult to understand by readers in our model iterations and propose improved models that guide readers’ understanding of the modeler’s story.

By improving the SEAM and i^* models, we aim to propose improvements to the SEAM and i^* notations. Therefore, we ask, “How should a SEAM / i^* designer design a SEAM / i^* notation so that modelers create models that readers understand?”. More broadly, we ask, “How should a designer design a notation that helps modelers create models that readers understand?”. We propose changes to the two notations and provide recommendations for improving other modeling notations as well. The recommendations are based on lessons learned from creating story-based models.

1.3 Architecture and IT Architecture

One of the most enduring metaphors in IT is architecture. The architecture of building has been used in the last thirty years as an analogy to the design of corporate IT systems (Zachman, 1987).

Architects use graphical models to communicate their conceptualization of a building with their clients and to facilitate the collaboration with model stakeholders (e.g., during meetings). For clients, architects show how the clients’ property fulfills the clients’ needs. For stakeholders (e.g., electricians, masons), architects show the characteristics of the property that interest them (such as the composition of walls and connectivity of pipes, circuits). (Sowa & Zachman, 1992) observe linguistic issues that appear when people from different disciplines work together. Specifically, the

work of the building architect includes a different vocabulary for each building level. Therefore, architects need to create models that clients and stakeholders, without an architecture background, understand. They connect the business world with the construction world.

(Zachman, 1987) uses the analogy between architecture and IT architecture to explain the work of IT architects. Similar to architects, IT architects use graphical models to communicate their conceptualization of an organization to their clients and to facilitate the collaboration between IT engineers (experts) and business people without an IT background (novices) during workshops. The story they show in models is how the strategy of the organization is formulated and followed, including the details about the information systems, and the roles and responsibilities of stakeholders. Therefore, IT architects need to create models that clients, without an IT background, understand. They connect the IT world with the business world.

The first three levels of Zachman's framework (Zachman, 2004), the owner's representation (model of the business = business description), the designer's representation (model of the information system = information system description) and the builder's representation (technology model = technology constrained description) are used to denote the role of the IT architect to combine the comprehension of business and IT stakeholders.

In this research, we use the building metaphor to explain how IT architects show stories in their IT models by analogy to how architects show stories in their construction models. IT architects are named modelers, graphical models are named models, the organization of concepts that the modeler conveys in his model is named story, and clients are considered readers of models. We enable modelers to create models so that readers can understand the modeler's story, by reducing the misalignment between the modeler's perception of the model and those of the readers.

Some model readers (e.g., with an IT background), familiar with the model's underlying (IT) modeling notation, can interpret the modeler's story. Other readers (e.g., without an IT background) are not familiar with the modeling notation and encounter greater difficulties interpreting this story. Our research concentrates on the second category of readers. We help IT modelers create models such that non-IT readers can understand the story. For instance, such models are used to summarize key decisions taken during workshops with IT and business stakeholders (e.g., how people work together with the IT – organize themselves) and enable the collaboration among them.

1.4 Thesis Organization

This thesis has the same structure as our research process. It is structured to reflect the search for improving the design of modeling notations in order for readers to understand the models created with these notations. Chapters 1, 2 and 3 represent the context of the thesis: we frame the problem of improving IT notations. Chapters 4 and 5 represent the conflict: we present our work on improving two IT notations: SEAM and i*. Chapter 6 represents the climax: we discuss our contributions to create improved models and design improved notations. Chapter 7 represents the closure: we draw conclusions and direct for future work. The context, conflict, climax and closure are the four story phases we use in this research. We apply them to mark the four thesis phases as described above.

The four thesis phases are presented in more detail for readers to understand the contents of each chapter as follows. In Chapter 1, we explain the context of the thesis, the main concepts used, our motivation for this research and the relation between architecture and IT architecture to justify the need to enable IT modelers to create IT models that non-IT readers understand. In Chapter 2, we describe the body of scientific literature that provides rigor to this research. The state of the art research includes IT modeling notations, evaluation frameworks for these notations, and model design frameworks. In Chapter 3, we explain the research method used in this thesis. In Chapter 4 and Chapter 5, we explain how we apply the research method to improve two modeling notations, SEAM and i^* , respectively. In Chapter 6, we discuss the implications of the current research and explain the theoretical principles and contributions of our research. In Chapter 7, we summarize the work and give directions for the continuation of the research. In the Appendix, we additionally include model iterations with SEAM and i^* , a case of modeling business strategy with SEAM, a case of modeling IT strategy with SEAM, a current and a proposed evaluation of SEAM using a notation evaluation framework, the Physics of Notations, and an initial presentation of other IT modeling notations.

Chapter 2 The State of the Art

We give an overview of three notations used in IT modeling: SEAM is an enterprise modeling notation used in business and IT alignment, i* is a goal modeling notation used in requirements engineering, and UML is a standardized notation used for the specification and design of information systems. Other IT modeling notations are briefly presented in Appendix 6. We then present four frameworks useful for evaluating IT notations. We conclude by presenting two model-design frameworks useful for modelers to model stories: the design analysis (based on Questions, Options, Criteria and Assessments) and the story phases (exemplified using the Context, Conflict, Climax and Closure).

2.1 IT Modeling Notations

We present several modeling notations used to create IT models. These modeling notations are used in the fields of business and IT modeling, requirements engineering, and software engineering by, e.g., business and IT consultants, enterprise architects, and systems designers. For each of the modeling notations, we explain what modelers use it for and present its meta-model, its set of graphical symbols and an example.

The currently-used IT modeling notations have origins in software engineering (e.g., i*, UML, BPMN). There are two main reasons that IT modelers use the notations inspired by software engineering: (1) They want to resemble the efforts of other engineers who draw graphical models, instead of writing programs; and (2) they want to communicate with others who do not have an IT background (Endres, 1996). Research efforts, such as Model Driven Architecture (Object-Management-Group, 2016), addressed the first need by creating tools that transform graphical models into executable code and vice-versa – this domain was named visual programming. Commonly, graphical models are used for the second need, despite the fact that less efforts were put into creating and improving IT models for non-IT readers.

IT modeling notations evolved in an ad-hoc manner. They were designed by software engineering experts, who were not experts in graphic design (Moody D. , 2009). Current IT modeling notations use simple and abstract graphical symbols, such as triangles, rectangles, parallelograms, cubes, diamonds, circles, ellipses and lines (straight, curved, with elbows, dashed or not, etc.) (Moody D. L., 2009). This constitutes an advantage in workshop settings, in which pen and paper are extensively used.

Little reflection has been made by the research community about the readers' understanding of modeling notations and their use. A notable exception is a notation named Visual Design Language (VDL), a part of a method called Solution-Based Modeling (SBM) (Goldstein & Alger, 1992). The authors hired a professional design-firm to help them develop a graphical notation that can be un-

derstood by both modelers and non-experts. They recommended that a notation be based on simplicity and ease of use, so that it can be used with only pen and paper as communication tools.

Most visual notations are evaluated based on their visual vocabulary or set of graphical symbols (Moody D. L., 2010), rather than the story that modelers convey in their models. The majority of IT modeling notations do not fit the purpose of serving as a communication media for non-IT people. IT modelers communicate with non-IT people by explaining collaboration on common projects via expert models. In short, IT modelers use IT notations to create models that they use to communicate (share their ideas and build relations) with non-IT people. The challenge is to improve existing IT notations, while maintaining their identity, so that non-IT people can understand the models created with these notations. This is necessary because IT notation designers want to promote their notation to modelers. They need to recognize the notation in the models that these modelers create for readers. We propose improved notations that have a similarly close identity (ideally the same) as the original ones, and are more understandable for readers. We do not design new notations.

2.1.1 Systemic Enterprise Architecture Method

The SEAM notation has been developed during the last twenty years. The below presentation of SEAM is based on the discussions with the members of the Laboratory of Systemic Modeling at EPFL, who developed the SEAM method (Wegmann A. , 2016), as well as on the descriptions of SEAM available in the publications that promote it ((Wegmann A. , 2016), (Wegmann, et al., 2013)).

SEAM is an enterprise modeling notation (Wegmann, Regev, Rychkova, Julia, & Perroud, 2007). Just like other enterprise modeling notations, such as ArchiMate (Lankhorst, H.A., & Jonkers, 2010) and DEMO (van Reijswoud, Mulder, & Dietz, 2001), SEAM is used in a variety of ways:

- In consulting, e.g., during workshops, to show stories involving stakeholders (business and IT representatives, clients). Workshops focus on early requirement phases in business strategy, business and IT alignment, enterprise architecture, and project management.
- In teaching, e.g., enterprise architecture, service-oriented architecture, requirements engineering, and business design for IT services.

SEAM modelers analyze and design business and IT systems that involve multiple stakeholders. They create models to show concrete organizational situations that are based on evidence from people. SEAM enables stakeholders to agree on what the issues are and what the solutions should be.

The main originality of SEAM, compared to other enterprise modeling methods, is its reliance on an explicit philosophical grounding (Regev, et al., 2013). This grounding, called the systemic modeling paradigm, is based on Systems Inquiry (Banathy & Jenlink, 2004). Systems Inquiry is composed of Systems Theory, Systems Philosophy, and Systems Methodology. Systems Theory is the view that the fragmented disciplines of science can be transcended by analyzing what is common among them. Systems Philosophy includes epistemology (where the model comes from), ontology (the structure of the model) and axiology (what elements to include in the model, or to exclude from it). Systems Methodology is the study and creation of methods for creating models. It includes the study of methods (their creation and improvement) and their practical use. These methods are used for the management of systems and related problems.

The SEAM epistemology relies on General Systems Thinking (Weinberg G. M., 1975), grounded cognition (Barsalou, 2008), and appreciative systems (Vickers, 1968). It is mainly based on the notion of a system, which is a relationship between an observer and an entity in an observed reality (Regev & Wegmann, 2005). Therefore, SEAM has an interpretative (Mintzberg, Ahlstrand, & J., 1998) or interpretive (Checkland & Holwell, 1998) epistemology. This means that SEAM modelers consider that each observer creates a unique, specialized view of the reality he observes, by interacting with the model and other observers. This viewpoint is reflected in the model with system boundaries.

The SEAM ontology provides constructs for behavior modeling. It is based on RM-ODP (Naumenko & Wegmann, 2002), goal-belief modeling (Regev, 2003), and value modeling (Golnam & Wegmann, 2013). Value modeling is based on marketing and quality theories, such as competitive advantage (Porter, 1980) and quality management (Hauser & Clausing, 1988).

The SEAM axiology consists in the choices made by SEAM modelers about what to include in their models, or to exclude from them. Choices are characterized by ethics (moral principles related to what to model) and aesthetics (usefulness, clarity and simplicity of the model).

Many tools can be used to create SEAM models. There are two custom SEAM tools: a stand-alone application (SeamCAD) and a web application (TradeYourMind). SeamCAD can edit all types of SEAM models. TradeYourMind proposes a specific way to combine the value with motivations modeling. It can be used to model business plans and IT systems requirements. Apart from the custom tools, SEAM modelers also use Microsoft PowerPoint / Apple Keynote and colored paper with Post-It notes to create models.

SEAM models are created using a hierarchy of systems (Figure 1). This hierarchy is called “Pure SEAM”. It can be used to model systems’ behavior (SEAM Behavior model), their motivations (SEAM Goal-Belief), or the value they create (SEAM Supplier-Adopter-Relationship). SEAM modelers want to understand the composites, marked graphically with “[c]”, in relation to the wholes, marked graphically with “[w]”, using this hierarchy. Each system can be analyzed as a whole, showing its externally visible characteristics, or as a composite, showing its interrelated parts. SEAM modelers refer to the fact that the whole is refined into a composite. The system shown as a whole includes services and their properties (for a Behavior model), or the goals and beliefs (for the Goal-Belief model), as perceived by an external observer. The system shown as a composite includes the sub-systems and the services that these sub-systems provide. The sub-systems participate in processes (for a Behavior model) and have goals and beliefs (for the Goal-Belief model).

A meta-model is a model of the model. It defines concepts, their relations and semantics. A model is an instance of a meta-model if it respects the structure defined by the meta-model. As seen before, the meta-model of PureSEAM includes: systems (wholes and composites), services (for wholes), properties (for wholes) and processes (for composites).

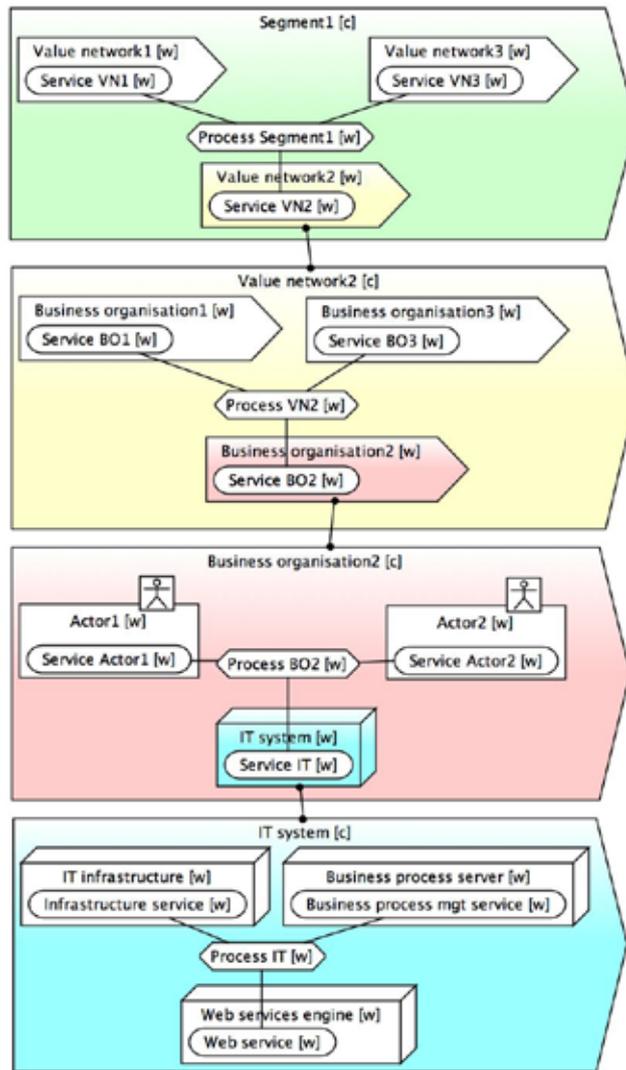


Figure 1: SEAM Behavior model showing a hierarchy of systems and their behavior

As SEAM is used to model services, the systems are called service systems. Figure 1 is an example of a Behavior model that shows service systems, services and processes. Service systems, as wholes, exhibit services. Service systems, as composites, exhibit processes. They are composed of sub-systems as wholes. These sub-systems provide services that are combined within processes. It is possible to relate business and IT needs with specifications. The model can be read either top-down or bottom-up. Service systems are represented as shapes with a surface that enables modelers to model either their composition (for a composite) or properties (for a whole). The refinement relationship is used to connect a system seen as a whole (e.g., Value network2 [w]) with the same seen as a composite (e.g., Value network2 [c]), or a service in a system seen as a whole (e.g., Service BO2 [w]) with the corresponding process in a system seen as a composite (e.g., Process BO2 [w]).

Once a system model is defined, it is possible to analyze each set of systems by using one of the three SEAM model types:

- *Goal-Belief* model: motivations modeling
- *Behavior* model: services and processes modeling
- *Supplier-Adopter-Relationship* model: components, features and values modeling

Goal-Belief model

The modeler's goal is to describe the relations between the goals and beliefs of model actors. He models the actors' beliefs and the reality aspects that drive them to take actions. The Goal-Belief model is based on the Appreciative System (Vickers, 1968) and includes three types of judgment: readiness to see (reality judgment), readiness to value (value judgment), and readiness to act (action judgment).

The meta-model of the SEAM Goal-Belief model is composed of: goal reduction, maintenance and achievement (action) goals, reality and value beliefs, and links (goal-goal reduction, belief-goal reduction, and goal-belief reduction).

The notation of the SEAM Goal-Belief model comprises the graphical symbols shown in Table 1, used to visually represent the components of the meta-model.



Table 1: Graphical symbols used for the SEAM Goal-Belief model

In Figure 2, the SEAM Goal-Belief example shows the connections between the values, realities, and actions of two persons. A student (John) and a company (Company X, represented by the HR Manager, Sam) participate in a university job fair. John's main goal is to obtain a job in a good company and Sam's is to recruit a talented student. John's beliefs can be related to the environment or the perceived reality (e.g., Company X is recruiting students with good programming skills) or his inner values (e.g., Company X is a good company, John has some basic programming skills that the company is looking for). By improving his programming skills, John influences the reality belief of Sam. Sam notices that John has attained the required skills and decides to recruit him. Stereotypes can be included, both on associations / links (maingoal, belief and goal) or in model elements (maingoal, value, reality and action).

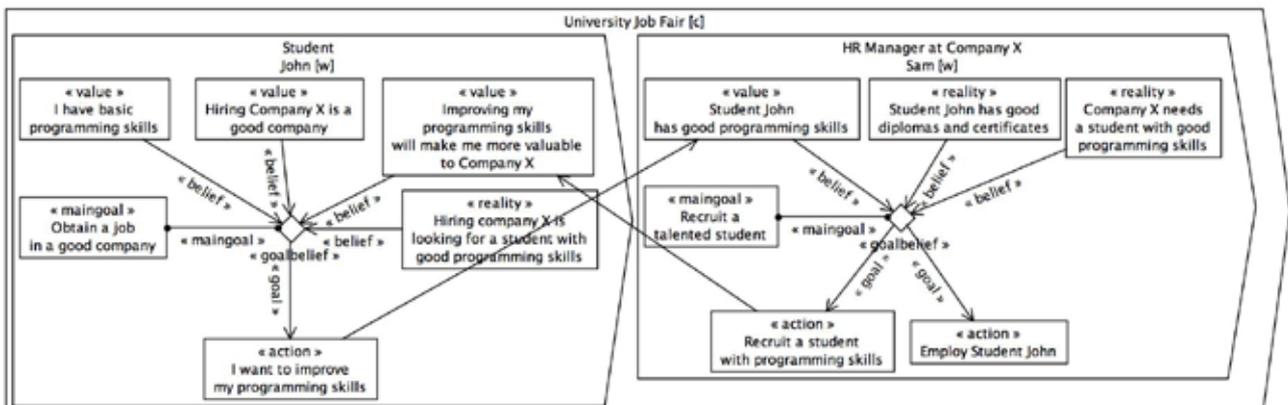


Figure 2: SEAM Goal-Belief model example

Supplier-Adopter-Relationship model

The modeler's goal is to describe the relation between a supplier and a customer (or adopter), by using components, features and values that are mapped together in a hierarchy and a matrix view (Figure 4). The supplier is a company. It is represented both as a composite (showing the components) and as a whole (showing the features). The adopter is represented on the right side (showing the values that he perceives). This model is created in order to illustrate the value that companies offer to their clients.

The meta-model is composed of: supplier-adopter-relationship reduction, component, feature, value, link between supplier-adopter-relationship and supplier value-network (as a whole and as a composite) and between adopters (as wholes).

The notation of the SEAM Supplier-Adopter-Relationship model comprises the graphical symbols shown in Table 3.



Table 3: Graphical symbols used for the SEAM Supplier-Adopter-Relationship model

In Figure 4, the SEAM-Supplier-Adopter-Relationship (SAR) example shows a supplier (company) providing service components (IT service, data maintenance, and incident management). These components are mapped to features (24 x 7 support, membership, and premium advantages). The features are perceived as values by the business partner (more customers and databases) and the individual partner (data management and e-mail support). The supplier is represented in the company value-network context both as a composite (bottom) and as a whole (top). The composite contains the IT department+ as a whole. In the matrix view, the relation between components, features and values is established with checked boxes. The two representations convey, in a different manner, the same story, i.e., the understanding of the correspondences between the service components (left) offered by the supplier, the features (middle), and the values (right) perceived by the two partners. An extension of this notation using a matrix view includes “++”, “+”, “-”, “--” to specify how the value is evaluated by the adopters, from strongly positively to strongly negatively. The positive and negative correlation is measured with three levels of green (+) and red (-) symbols. The matrix view of the SAR complements the SAR model (the latter is not supposed to show the details).

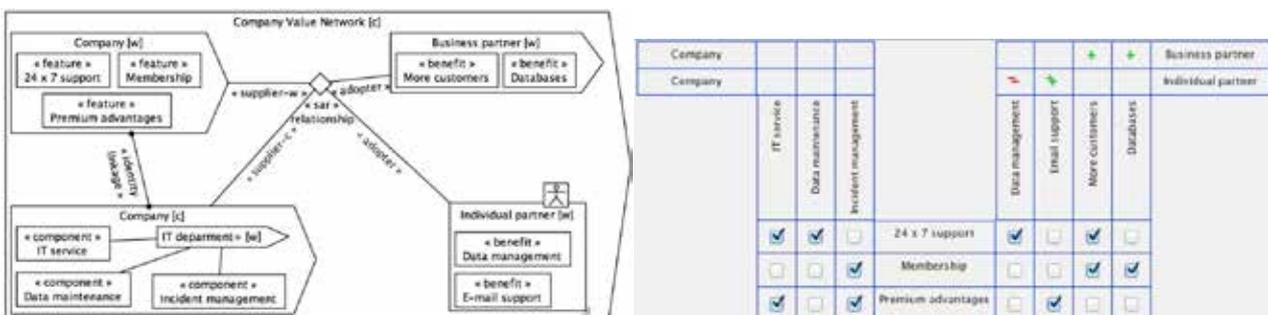


Figure 4: SEAM-Supplier-Adopter-Relationship model and matrix example

SEAM Notational Principles

The SEAM notation is based on UML (Fowler, 2004) and Catalysis (D'Souza & Wills, 1999). It uses some of the principles defined in software engineering, but proposes a way to create models that is different from other IT notations (Figure 5).

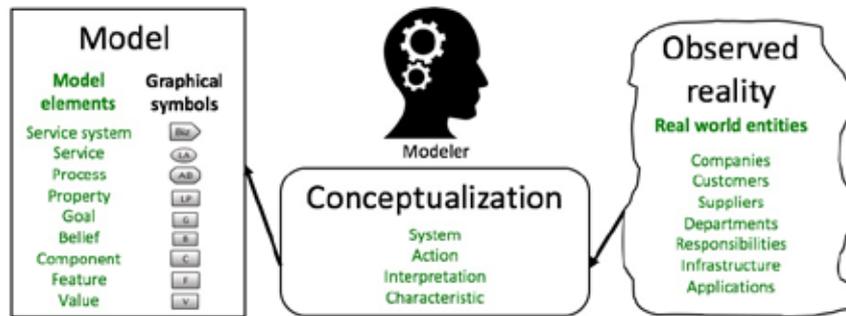


Figure 5: SEAM modeler's conceptualization

The SEAM notation is based on the following principles:

- Generic system modeling with specific ontological relation to reality.** The main originality of SEAM is its epistemological foundation. To understand SEAM modeling, it is useful to consider the observed reality in which real world entities are perceived (Figure 5). These entities (e.g., companies, customers, suppliers) are perceived by the modeler through his conceptualization. They are represented in the model with graphical symbols, as model elements (e.g., service system, service, process, property, goal, belief). As SEAM is systemic, the SEAM modeler's conceptualization includes systems. All systems are modeled in a similar way: they have characteristics and perform actions in their environment. The human systems have interpretations. A challenge with SEAM is to help the modeler and the readers know precisely what is described. For example, the modeler can describe the components of an IT system that provides a service; or he can describe the collaboration of companies who provide a service. The way to describe how the components of the IT system work together and the way to describe how the companies work together are the same. For this reason, the notation is both generic and specific. Generic systems modeling enables the modeler to represent systems, services and processes in a similar way. The specificity of SEAM modeling enables the modeler to use different icons to represent different kinds of systems (e.g., a company, a department, an IT system, a server, a data center), which is a specific ontological relation to reality. Practically, the systems shown in Figure 1 all appear different, because they represent different kinds of entities (company, department, team, IT infrastructure). But the modeling constructs used to represent motivations (Figure 2), behavior (Figure 3) and value (Figure 4) are the same, regardless of the represented systems (i.e., regardless in which context they exist). Hence, a business process or an IT process will have the same graphical symbol. An IT infrastructure, however, will have a different graphical symbol than a company.
- 2D graphical symbols to support refinements.** Almost all SEAM graphical elements have two dimensional (2D) shapes; this is useful to show the whole-composite relationship. A system, as a whole, has properties (e.g., in a Behavior model). A system, as a composite, has component sub-systems as wholes. The whole-composite relationship can be applied to ser-

vices, processes and properties. A service, as a composite, includes component sub-services. A process, as a composite, includes component sub-processes. A property, as a composite, includes sub-properties. The “whole” notation is designed to support the refinement of wholes into composites of all types of graphical elements. For practical reasons, SEAM modelers do not show all refinements in order for readers not to get lost in details. For instance, associations cannot be decomposed due to their one-dimensional (1D) support. This is a limitation that could be addressed in a future work.

- **Rectangles for properties.** Systems exhibit properties. For example, a Behavior model of a system shows how a service changes state when the service occurs; it can also show input and output properties. A Goal-Belief model shows predicates that capture the system’s reasoning. These predicates are stored in properties. All properties are represented as 2D rectangles. The type of information stored in the property can be indicated with a stereotype. This is convenient for relating models. For example, a goal in a Goal-Belief model can be a value in a Supplier-Adopter-Relationship model. For this reason, the shapes of properties are identical in all models (shown as rectangles).
- **N-ary associations.** These associations are useful during a design process. They express a relation between multiple elements, without over-specifying this relation. All n-ary associations are represented with either four- or a six-sided diamonds. The semantics of the n-ary associations depends on the model in which it is used.
- **Ovals for services.** Behavior is shown with ovals. A behavior is defined as a model element that represents a change of a state in time. A process is not a behavior because it connects services. The process is included in the model only to show the relations between the services. As such, the process does not have pre- and post-conditions. Conceptually, a process is close to the n-ary association because it relates services. This is why, graphically, a process is represented with a hexagon (6 sides) whereas the n-ary association is represented with a diamond (4 sides).
- **Relation with trade notations.** SEAM adopted graphical symbols used in the trade domain. For systems, SEAM uses graphical symbols either from business representation (e.g., Porter’s value chain) or from UML (e.g., actors, components, applications). The behavioral elements (services) are oval, as in UML (use cases, activities and possibly states). The properties are rectangles, as in the UML class diagram. SEAM uses these graphical symbols because they are easily understood by business and IT people. If SEAM had been developed for people working in manufacturing, for instance, then the block arrow could have been used for the behavioral element because processes in quality documentation in manufacturing are represented with arrows.
- **Only one model instead of more.** UML defines fourteen model types. There is one model for each category of information. For example, a class diagram shows properties, an activity diagram shows behavior, the systems are shown in a use-case diagram and/or in a component diagram. Some of the relations between these diagrams are captured in Object Constraint Language (OCL) formal models. In contrast with UML, in SEAM, the system, its behavior and the related properties are shown in only one model (SEAM Behavior). It is then possible to see in which context a behavior is defined and what properties are specified for that behavior. Therefore, SEAM modelers need to show more aggregate information in one model because more types of information are shown together.

- **Goal-Belief, Behavior and Supplier-Adopter-Relationship models.** SEAM proposes three types of models that are defined for a specific purpose. The Goal-Belief model is specific to SEAM – it is used to model actor’s motivations. UML currently has the Business Motivation Model for this purpose. Modeling with SEAM Behavior model is similar to UML behavior modeling. The Supplier-Adopter-Relationship model is a special type of model that includes a Behavior model (that includes the features of a service and the components necessary for these features), as well as a subset of a Goal-Belief model (that includes the values that the features offer for a customer). Therefore, SEAM proposes models that are not based on types of elements (as UML does). The three types of SEAM models serve a targeted purpose: to understand aspects of the behavior, aspects of the motivation, or how behavior and motivation are combined in order to explain value creation, respectively.
- **Points of view.** The main concept of a system, representing the relationship between an entity and its environment (observed reality: (Weinberg G. M., 1975)), is used not only in the SEAM epistemology, but also in the SEAM models themselves, where explicit points of view are shown.

SEAM notation examples

SEAM models are often created in workshops (with flipcharts, Post-It notes and markers). Models are drawn by hand to document the results and the stakeholders’ emotions. It is frequent that variations of the notation are used.



Figure 6: SEAM model of SITRA using the workshop notation

In Figure 6, we illustrate one variation of the SEAM workshop notation. It is different from the SEAM Behavior model described previously, but it uses the same principles. It was used in a four-day workshop with ten participants working for a touristic organization named SITRA. The goal of the workshop was to define an IT strategy (left) that supports a business strategy (middle) for segments of clients (right). In the middle of the model, the collaboration between various companies was analyzed. The model represents the three levels of systems (from left to right: IT, companies and business). The tapes and the border of the craft paper shows the system boundaries. With this notation, the notion of system is hidden, but the boundaries are maintained.

The workshop participants did not need to understand the system concept (but the workshop animator did). The ovals (services) and diamonds (processes) were preserved. The color-coding enabled workshop stakeholders to understand what was used at each level (in each context / boundary). Some Post-It notes are used to represent goals and beliefs. Some logos and pictures were used to make the model more concrete. The notation was developed to animate the workshop without explaining the SEAM principles, while still using them.

A mapping between the workshop and the academic notations can be established using the same colors and shapes (Figure 7): properties are marked with orange rectangles in the tourist boundary, the services provided by the tourist office value network are marked with blue ovals, the organizations of the tourist office value network are marked with box arrows, and the services provided by the SITRA value network are marked with blue ovals. The complete mapping including all actors and details (in the tourist office value network and in the SITRA value network) is included in Figure 100 in Appendix 3.

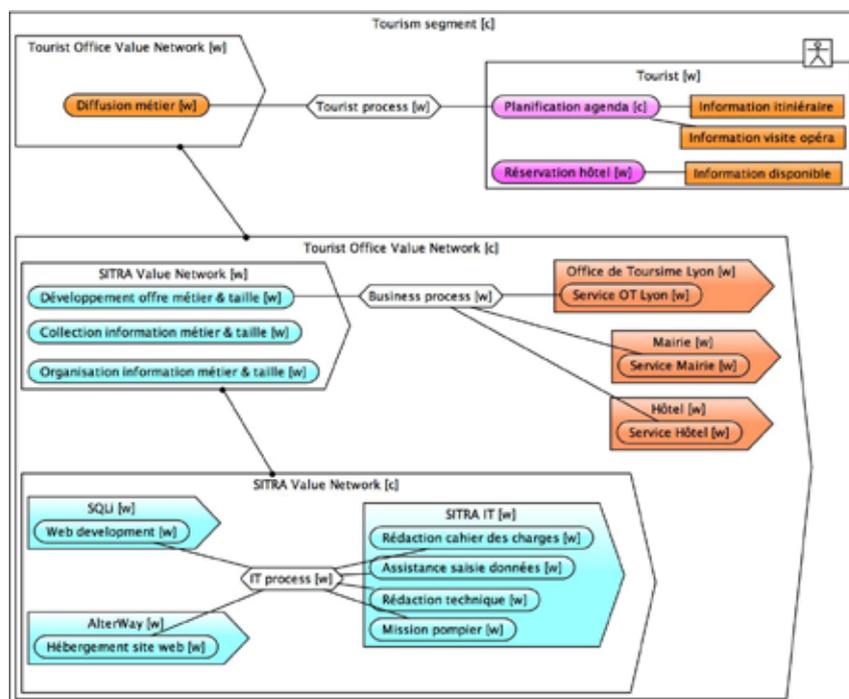


Figure 7: SEAM model of SITRA using the academic notation

In Figure 8, an online version of SEAM implemented for TradeYourMind (TYM) is shown (Etzlinger, Castori, & Wegmann, 2016). TYM is an online platform that helps entrepreneurs investigate business ideas. In TYM, the hierarchical structure of SEAM is represented using colors and menus. The model created in TYM corresponds to the Supplier-Adopter-Relationship (SAR) (mapping shown in Figure 9), a partial Behavior model, and a Goal-Belief model. One of the benefits of TYM is that it provides an integrated and guided way to build SEAM models.

The graphical structure of the TYM model is inspired by the Business Model Generation (BMG) (Osterwalder & Pigneur, 2016), a competing business modeling technique. Similarities include the flow of information from key partners (left) to customers (right). The SEAM canvas includes the regulation authorities and the competition whereas BMG does not. The BMG canvas enables modelers to analyze key activities and resources, value propositions, customer relationships, and chan-

nels. The SEAM canvas enables modelers to analyze components (similar to resources), features and benefits (similar to value propositions). TYM also includes the financial view that contains revenue and cost factors of the organization but this is modeled separately (not in the canvas).

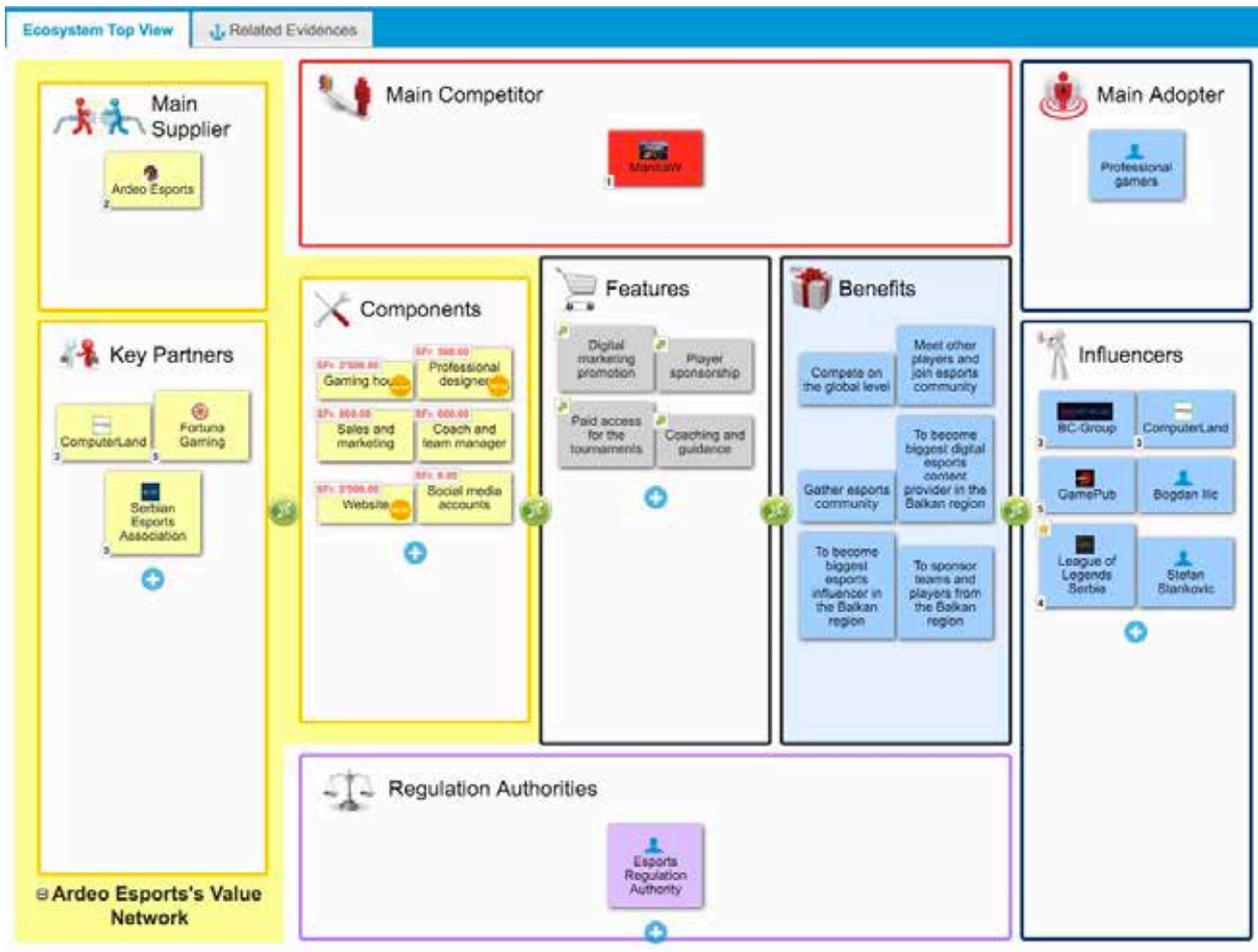


Figure 8: SEAM model in TradeYourMind done with the online notation – source: (Etzlinger, Castori, & Wegmann, 2016)

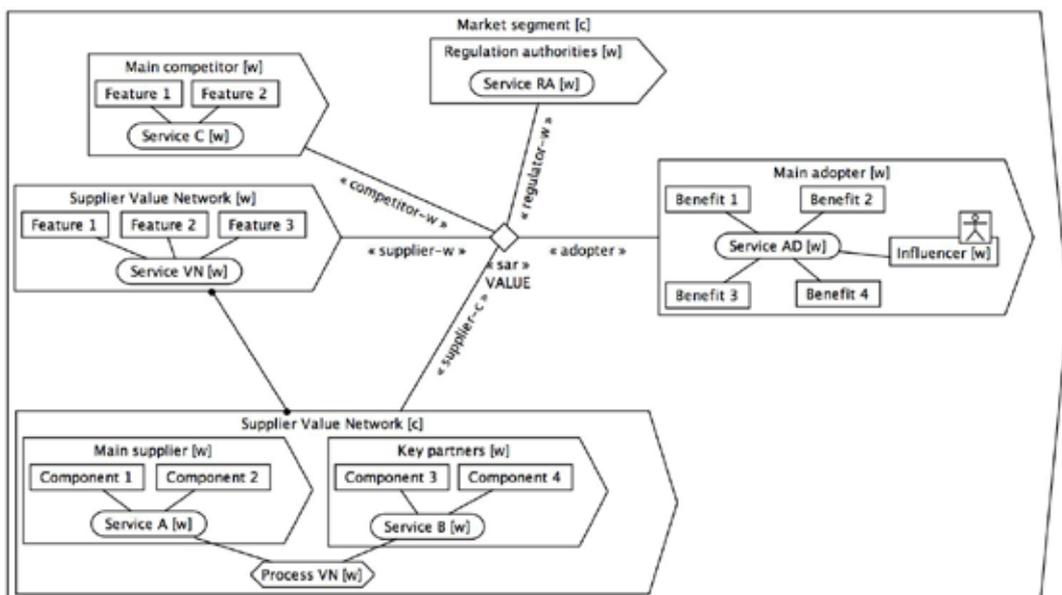


Figure 9: SEAM model of TradeYourMind using the academic notation

In this research, we improve SEAM models starting from the academic notation, not the workshop or online notations, because it represents the reference for SEAM modeling in terms of modeling principles, and because the modeler would ideally like workshop participants to understand the SEAM academic notation.

2.1.2 i*

The below presentation of the i* notation is based on several publications describing it, such as: (Yu E. S., 1997), (Yu & Mylopoulos, 1994) and (Yu E. , 2001).

i* is one of the most widely used goal modeling notations used in requirements engineering. It was previously applied to business process modeling and redesign, as well as to software process modeling. It is useful to tell goal-oriented stories.

i* consists of two models:

- *Strategic Dependency* (SD) model: used to “describe the dependency relationships among actors in an organizational context” (Yu E. S., 1997)
- *Strategic Rationale* (SR) model: used to “describe the rationale leading to a decision of actors by showing actors’ interests and concerns, and how they might be addressed by various configurations of systems and environments” (Yu E. S., 1997).

i* modelers communicate with business stakeholders in early-requirements project-phases. i* modeling results in high-level business models, without going into details about the processes and technologies. Modelers focus on understanding both the organizational context and rationales (the “Whys”) that lead to systems’ requirements.

The central concept in i* is the intentional actor. Actors have properties such as goals, beliefs, abilities and commitments, and depend on each other for goals to be achieved, tasks to be performed, and resources to be furnished, via strategic relations (Yu E. S., 1997). An actor (dependum) may be able to form goals that are difficult to achieve alone, by depending on other actors (dependees). The i* model helps to model stakeholders’ interests when they seek opportunities or changes.

The i* notation (Yu E. , 2001) is presented in Figure 10 and includes:

- *Actors*: are entities that perform actions to achieve goals. Agents, roles and positions are specialized views of the actors.
- *Goals*: are intentional desires of an actor that can be achieved in many ways.
- *Softgoals*: are qualitative criteria. From the softgoals, one can tell why one alternative may be chosen over others. Softgoals draw on the concept of satisfying, which refers to finding solutions that are “good enough”.
- *Tasks*: are specific ways to accomplish goals.
- *Resources*: can be physical or informational.
- *Beliefs*: are conditions about the world that the actors think are true.
- *Consisting of*: is represented via task-decomposition links.
- *Task-decomposition link*: provides a hierarchical description of intentional elements that make up a routine.

- *Means-end link*: shows when a goal can be met. Means-end links provide understanding about why an actor would engage in some tasks, pursue a goal, need a resource, or want a softgoal.

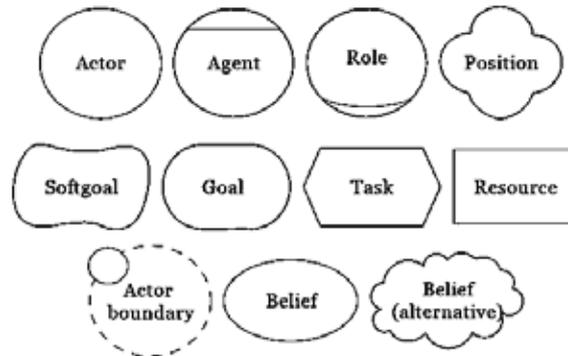


Figure 10: i* notation – source: (Yu E. , University of Toronto, 2001)

The field of requirements engineering (RE) can be divided into functional requirements (FR), that centers on goal analysis, and non-functional requirements (NFR), that centers on softgoal analysis. Goals have been used to connect requirements with design. When designing new systems (e.g., to identify the relevant characteristics the system should support) or comparing alternatives (e.g., two systems), modelers need to analyze the interrelations between goals and softgoals (Mylopoulous, Chung, & Nixon, 1992). Goals are useful for objectively measurable scenarios, whereas softgoals are useful for ill-defined goals and their interdependencies, for instance when no clear criteria that satisfy them exists. i* modelers consider that softgoals are “satisfied” when there is sufficient positive evidence for their claim. One softgoal can support (this is marked with a “+” sign, and is interpreted as “positive influence”) or can conflict another (this is marked with a “-“ sign, and is interpreted as “negative influence”). In order to show when a goal is satisfied, i* modelers select a partial set of softgoals that collectively satisfy it. Satisfying all softgoals of a system design might be impossible because of conflicts. The decomposition of goals into softgoals can be project-specific or task-specific (Mylopoulous J. , Chung, Liao, & Wang, 2001).

In order to create an i* model, modelers need to know what tasks, goals and resources are required, and what softgoals are pertinent. In analysis, alternatives are evaluated with respect to goals. In design, goals are used to generate potential solutions systematically. The starting point is to express the customer’s wishes about what the system should do. However, these are often ambiguous, incomplete, inconsistent and usually expressed informally. i* aims to be a systematic framework for modelers to understand what users want and for users to understand what the system does. When refining the model, i* modelers perform the following phases: goal analysis, softgoal analysis, softgoal correlation analysis, goal correlation analysis, and evaluation of the design (Yu E. , 1995).

As opposed to other requirements engineering notations, i* focuses on the “Why?” rather than the “What?”. This is useful for the cooperation between the actors as well as for the design of the system.

The models presented in Figure 11 and Figure 12 consider a computer-based meeting scheduler used for supporting the setting up of meetings between a meeting initiator and meeting participants. When creating an i* model, the modeler might ask himself:

1. What are the needs for scheduling meetings?
2. What do the meeting participants need to provide?
3. Why is a scheduler desired by the two parties?

Figure 11 shows the strategic dependency model of meeting scheduling with a scheduler. This model can be interpreted by an i* modeler (Yu E. S., 1997) as follows: “The model shows the scheduling of a meeting with a computer-based meeting scheduler. The meeting initiator delegates much of the work of meeting scheduling to the meeting scheduler. The initiator does not need to be bothered with collecting agreements about proposed dates from participants. The meeting scheduler determines what are the acceptable dates, given the availability information. The meeting initiator does not care how the scheduler does this, as long as the acceptable dates are found. The scheduler expects the meeting initiator to enter the date range by following a specific procedure modeled with a task dependency. The initiator needs participants to attend the meeting, in order to attain a goal.”

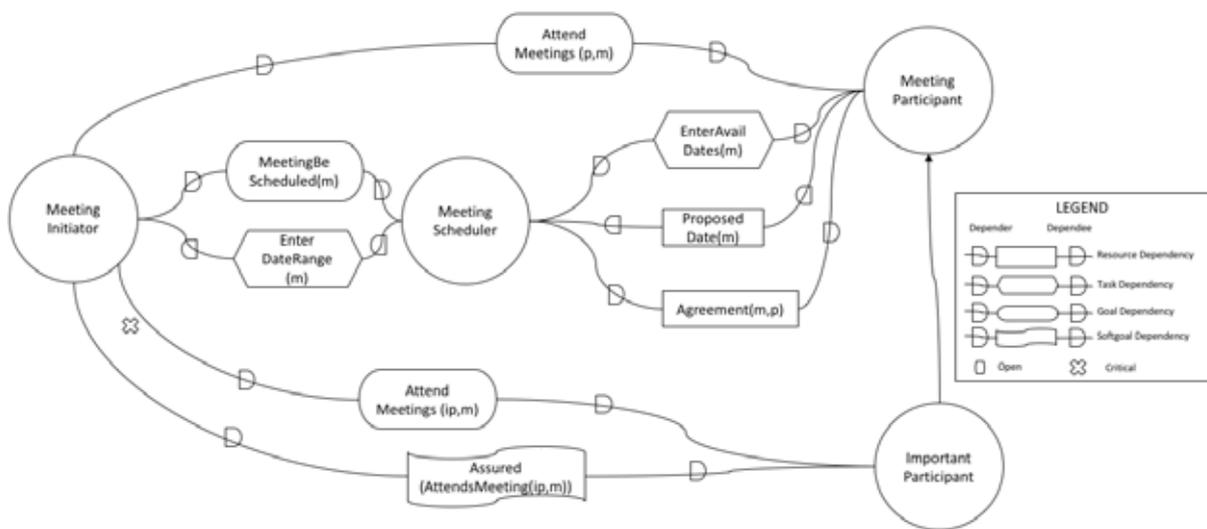


Figure 11: i* Strategic Dependency (SD) model example – source: (Yu E. S., 1997)

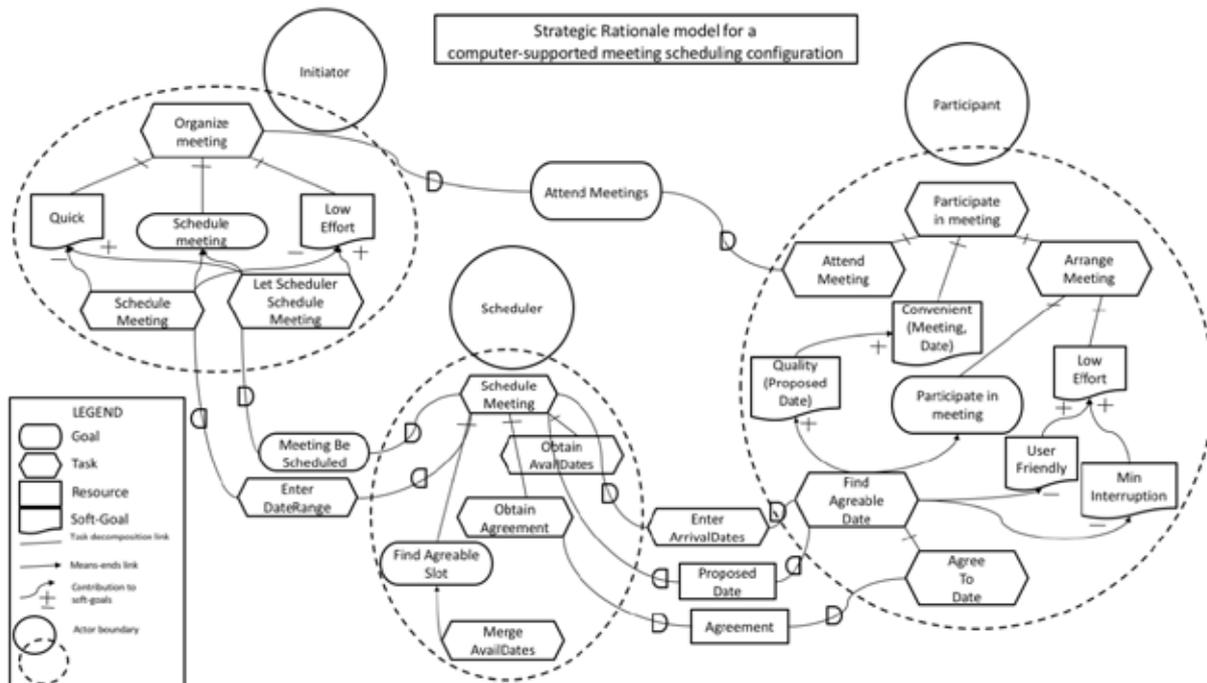


Figure 12: i* Strategic Rationale (SR) model example – source: (Yu E. S., 1997)

Figure 12 shows the strategic rationale model of the meeting scheduling with a scheduler. This model can be interpreted by an i* modeler (Yu E. S., 1997) as follows: “The model provides a way of modeling stakeholders interests, and how they might be met, and the stakeholders’ evaluation of various alternatives with respect to their interests. The availability information in the form of exclusion sets and preferred sets is collected so as to minimize the number of rounds and thus to minimize interruption to participants.” The three actors exhibit goals, tasks, resources and soft-goals that are linked with task decomposition and means-end links.

The above i* models are useful to illustrate the kind of modeling and reasoning support that is performed by i* modelers during early phase requirements engineering. A precise specification of all interactions would require a much larger set of concepts and relationships. These can be performed using structuring mechanisms, such as classification, generalization and aggregation. By combining the two models, the i* modeler can characterize the relationships among actors at the intentional level in the strategic dependency model (without knowing the actor’s internal intentional flows in detail) and, when reasoning about alternative configurations, he can make the goals and criteria for deliberations more explicit in the strategic rationale model.

2.1.3 Unified Modeling Language

The below presentation of the Unified Modeling Language (UML) notation is based on numerous publications describing it, such as: (Unified Modeling Language, 2016), (Fowler, 2004), (Booch, Rumbaugh, & Jacobson, 2005), (Booch, 1994), (Jacobson, Christerson, Johnsson, & Overgaard, 1997), and (Larman, 2005).

UML is managed and was created, by the Object Management Group (Unified Modeling Language, 2016), an international standards provider. Since 1997, UML became a software engineering standard. UML was designed for the specification and design of information systems. Both UML and SEAM-Behavior model are used to design software systems.

UML modelers specify, visualize, modify, construct and document the artifacts of an object-oriented software-intensive system under development (Fowler, 2004). Models offer a standard way to visualize a system’s architectural blueprints, including elements such as: activities, actors, business processes, database schemas (logical), components, programming language statements, and reusable software components. Modelers combine techniques from data modeling (entity relations diagrams), business modeling (work flows), object modeling, and component modeling.

The UML meta-model defines the structure that all UML models must have (Booch, Rumbaugh, & Jacobson, 2005). Meta-model elements are spread across model types, which define overlapping views of the meta-model (Figure 13). Models are named diagrams in UML. They consist in: class, attribute, instance, association, operation, component, collaboration, role, stereotype, etc.

The UML notation is synthesized from the notations of the Booch method (Booch, 1994), the object-modeling technique (Rumbaugh, 1991), and object-oriented software engineering (Jacobson, Christerson, Johnsson, & Overgaard, 1997) by combining them into a single modeling language.

A UML model is a partial graphical representation (or view) of a system under design, implementation, or already in existence (Unified Modeling Language, 2016). The UML model of the system

might contain other documentation; such as use cases written as template texts. The model's kind is defined by the primary graphical symbols shown in the model.

UML specification does not prevent mixing different kinds of models, e.g., combine structural and behavioral graphical elements to show a state machine nested inside a use case. Consequently, the boundaries between the various kinds of models are not strict. However, some UML tools restrict the set of available graphical elements which could be used when working on a specific type of model.

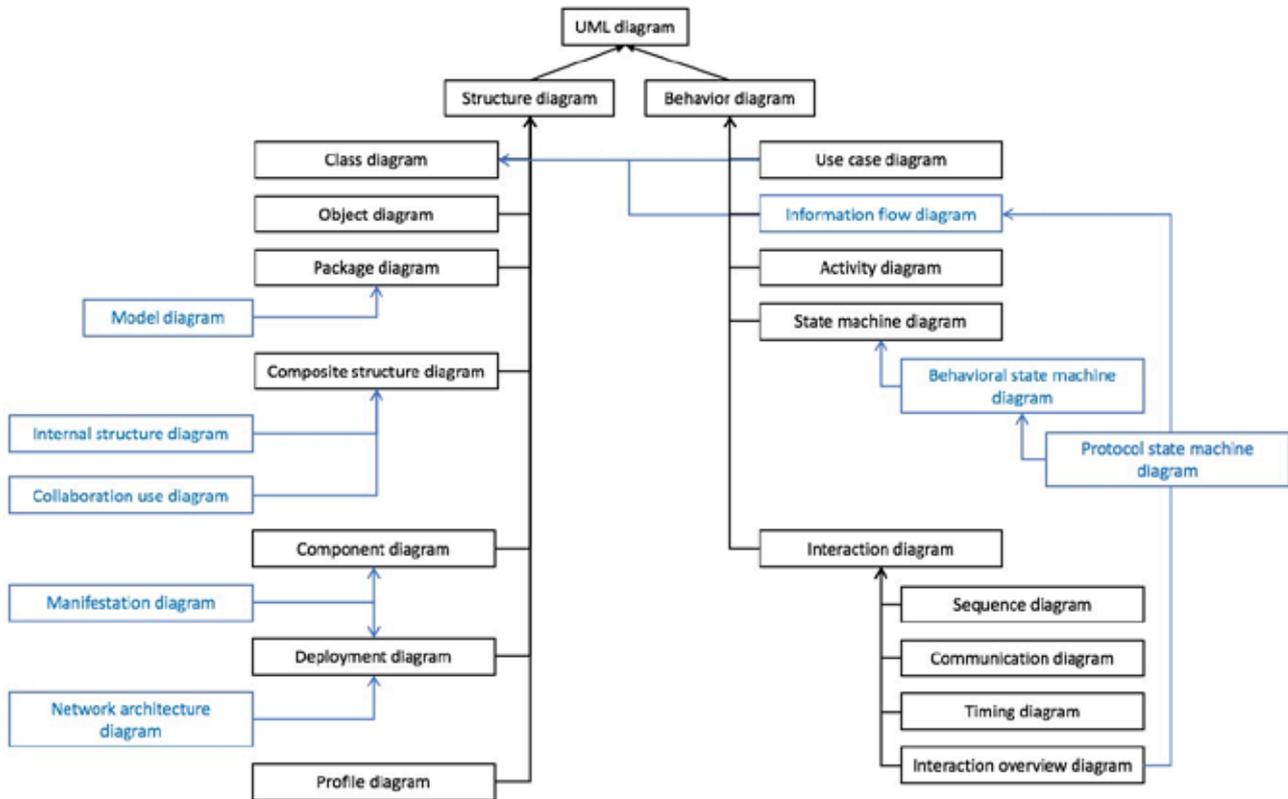


Figure 13: Unified Modeling Language model types – source: (Unified Modeling Language, 2016)

The following synthesis of UML model types is based on (Unified Modeling Language, 2016).

Structure diagrams

Structure diagrams (Figure 13, left branch) include the elements that must be present in the system under consideration. They emphasize the static structure of the system using objects, attributes, operations and relations. They are used extensively in documenting the software architecture of software systems. Structure diagrams are:

- *Class diagram*: describes the structure of a system by showing the system's classes, their attributes, and the relations among the classes.
- *Component diagram*: describes how a software system is split up into components and shows the dependencies among these components.
- *Composite structure diagram*: describes the internal structure of a class and the collaborations that this structure makes possible.
- *Deployment diagram*: describes the hardware used in system implementations and the execution environments and artifacts deployed on the hardware.

- *Object diagram*: shows a complete or partial view of the structure of an example-modelled system at a specific time.
- *Package diagram*: describes how a system is split up into logical groupings by showing the dependencies among these groupings.
- *Profile diagram*: operates at the meta-model level and shows stereotypes as classes with the <<stereotype>> stereotype, and profiles as packages with the <<profile>> stereotype.

Behavior diagrams

Behavior diagrams (Figure 13, right branch) show the dynamic behavior of the elements in a system, which can be described as a series of changes to the system that occur over time. They show collaborations among elements and changes to their internal states. Behavioral diagrams emphasize what must happen in the system being modelled. They are used extensively to describe the way in which the system functions. A behavior diagram denotes action, event, message and state to depict how UML graphical symbols interact in the model. Behavior diagrams are:

- *Activity diagram*: describes the business and operational workflows of components in a system; it shows the overall flow of control (e.g., via information).
- *State machine diagram*: describes the states and state transitions of the system; it shows the context, focusing on the data.
- *Use case diagram*: describes the functionality provided by a system in terms of actors, their goals represented as use cases, and any dependencies among use cases.
- *Interaction diagram*: describes the sequence of events, the communication, the timing or the flow of control between elements

UML relies on a universal ontology, with both time independent models (structure) and time dependent (behavior) models. Time, ordering and multiplicity constraints are well specified in UML, although scattered in several models, patterns and other artifacts. In UML models, objects are not placed in context – the context is implicit (not shown), not explicit (shown) as in SEAM models, in which the role of systems’ boundaries is considered by SEAM modelers as critical. The boundary in UML is not considered an important concept, so it can be hidden. Graphical symbols cannot be changed, because they are concepts non-dependent on the context. In the UML use case diagram, it is possible to hide the IT system’s boundary. The underlying principle that explains that the IT system’s boundary is hidden in UML is called Occam’s razor principle. This principle expresses that a succinct model is better than a complex / complicated one. Similar to SEAM, UML elements are intended to be drawn on 2-dimensional surfaces. The focus in UML is to define detailed level IT specifications, whereas in SEAM it is to link business and IT requirements.

In Figure 14, we show one example of an UML activity model (Fowler, 2004). According to (Fowler, 2004), it shows a payment activity that returns either “succeeded” or “failed”. Payment type can be of three kinds: by credit card, by check, or by invoice. For credit, if the authorization of the credit card was ok then the transaction has succeeded. If not, it has failed. For check, the situation is the same. For invoice, if the customer is regular and there is a payment history, then the transaction is successful. If the customer is regular and the value is lower than \$1’000 or if the payment history check has failed, then a prepayment request is necessary. Other possibilities are shown using directed lines.

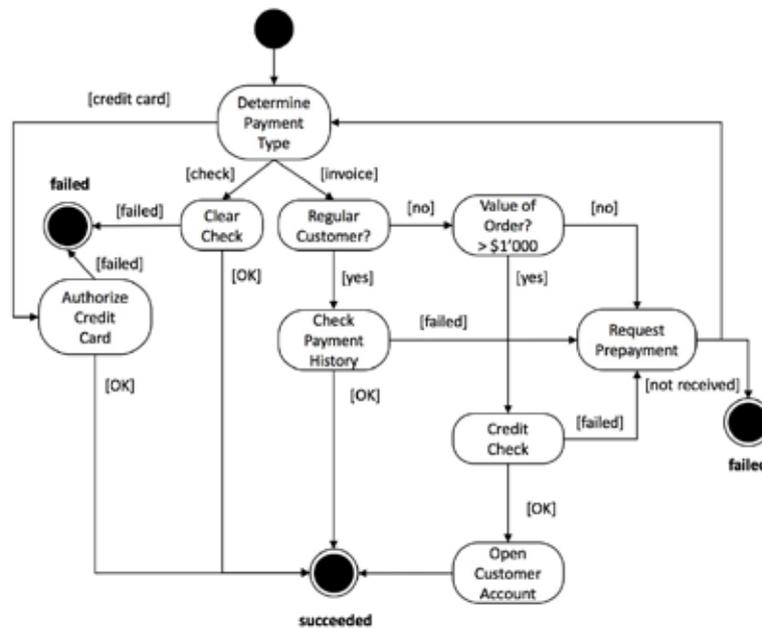


Figure 14: Unified Modeling Language activity diagram notation and model example – source: (Fowler, 2004)

2.1.4 Concluding Remarks

In Chapter 2.1, three examples of commonly used IT modeling notations have been presented. Most of these notations include simple, abstract shapes (e.g., ovals, rectangles, lines). They are based on the experience and good practices developed by IT modelers. The experience and the practices are captured by means of patterns that are reused from project to project (Moody D. , 2009). It becomes challenging to promote, teach and apply these notations to readers unfamiliar with them. This is because readers encounter difficulties in identifying: (1) what do the elements represent (relation with reality), (2) what is the problem (rationale), and (3) what happens in the model (story). Therefore, modelers lack the representation of the relation to reality, the rationale and the story in their models.

There are several aspects related to the relations among the above-mentioned modeling notations that are worth mentioning. What stands out in SEAM compared with other notations is the entangled connection between how knowledge is formed (modeling) and the set of elements used in describing this knowledge (the model). With a limited set of graphical elements, SEAM enables modelers to express their perceptions of the world using elements (photos, icons, terminology) that they know. These elements reflect their understanding of the world, as it is perceived and constructed by the modeler. By opposition, other notations rely on a predefined set of graphical elements supposed to encompass every aspect of reality.

The SEAM notation is inspired by the Unified Modeling Language (UML). A partial set of the SEAM notation is similar to UML. SEAM takes the UML notation and proposes one kind of diagram that includes a subset of the element kinds found in some UML diagrams. A SEAM diagram is a combination of the UML deployment diagram, use case diagram, and class diagram. It includes composition relations. The emphasis is set on hierarchical concepts. SEAM can be used to complement the UML and the Business Process Modeling Notation (BPMN) for system pre-design. SEAM designers do not aim to show low-level design, but to delimit the problem and analyze stakeholders' perceptions or viewpoints.

2.2 Evaluation of Modeling Notations

Notations are used in all areas and all levels of IT practice, from strategic planning to the design of integrated circuits. They play a critical role in communicating with end users and customers, because they convey information more effectively to non-technical people than text. Therefore, the desirable goals for modelers of IT notations should be to “maximize precision, expressiveness and parsimony” (Moody D. L., 2010). These attributes should enhance the communication among business and IT stakeholders, but are difficult to objectively measure. In this regard, four main notation-evaluation frameworks have been proposed in the scientific literature: semiotic quality framework, seven process modeling guidelines, cognitive dimensions of notations, and physics of notations.

2.2.1 Semiotic Quality Framework

The semiotic quality (SEQUAL) framework is a reference model used to evaluate the quality of models (Krogstie J. , 2006). It is also known as the “top-down” framework (Krogstie J. S., 2003). The framework considers “quality aspects based on the relations between a model, a body of knowledge, a domain, a modeling method, and the activities of learning, taking action, and modeling” (Krogstie, 2006) (Figure 15). It is, therefore, a descriptive rather than analytical evaluation framework. SEQUAL was applied to evaluate process models by (Moody & al., 2002).

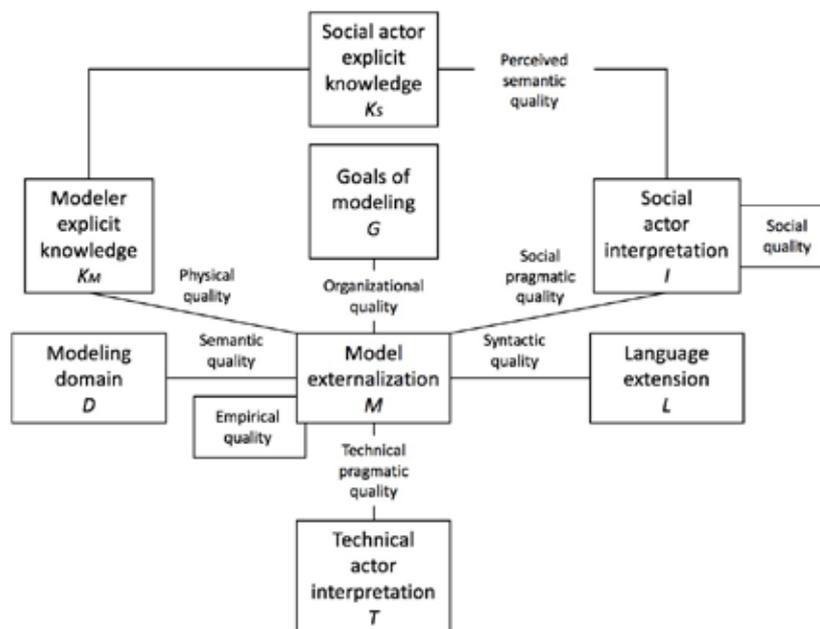


Figure 15: Semiotic Quality framework – source: (Krogstie J. S., 2003)

SEQUAL considers eight elements (Krogstie J. , 2012):

- *A*: Actors create (parts of) the model (e.g., persons or tools)
- *L*: What can be expressed in the modeling notation
- *M*: What is expressed in the model
- *D*: What can be expressed about the domain (area of interest)
- *K*: The explicit knowledge of the participating persons
- *I*: What the persons in the audience interpret from the model
- *T*: What relevant tools interpret from the model

- *G*: The goals of the modeling

SEQUAL can be used to evaluate the “empirical quality” of a model. This quality refers to the question “Is the model easily understandable?”. Empirical quality considers the variety of graphical elements that readers can distinguish, the errors when these elements are created or interpreted, and coding (shapes of boxes). (Krogstie J. , 2012) mention factors that have an impact on visual comprehension, e.g., size, solidity, foreground/background differences, color, change of state, position.

2.2.2 Seven Process Modeling Guidelines

The seven process modelling guidelines (7PMG) studies the relations between model structure, on the one hand, and error probability and understanding, on the other hand (Mendling, Reijers, & van der Aalst, 2009). 7PMG includes desirable properties when changing a process model to a behavior, more understandable, one. 7PMG provides a set of recommendations (or heuristics) on how to create and improve process models. They are helpful in guiding modelers to improve the quality of their models. Improvements take two forms: (1) become comprehensible to readers, and (2) contain less syntactical errors.

Each of the seven guidelines (Table 4) builds on empirical and theoretical insights, yet they are formulated intuitively for practitioners. Each suggests possible improvements for a process model together with alternatives of a set of preferred behavior-equivalent representations. Some guidelines make objectively measurable suggestions: “use one start and one end event” or “decompose the model if it has more than 50 elements”, whereas others are less well-specified “use as few elements in the model as possible” or “model as structured as possible”.

Index	Guideline
1	“Use as few elements in the model as possible.”
2	“Minimize the routing paths per element.”
3	“Use one start and one end event.”
4	“Model as structured as possible.”
5	“Avoid OR routing elements.”
6	“Use verb-object activity labels.”
7	“Decompose the model if it has more than 50 elements.”

Table 4: Seven Process Modeling Guidelines – source: (Mendling, Reijers, & van der Aalst, 2009)

2.2.3 Cognitive Dimensions of Notations

The cognitive dimensions of notations (CDNs) is an approach to analyzing the usability of information artifacts, often software systems or daily objects (Green, Blandford, Church, Roast, & Clarke, 2006). CDNs can be applied to discover aspects related to problems that are not analyzed using conventional techniques from ergonomics or human-computer interaction (Green & Petre, 1996). The framework of cognitive dimensions is applicable to notations (e.g., related to music, dance, Morse code, programming languages), and to information-handling devices and tools (spreadsheets, databases, word-processors). The analysis of usability of notations can be applied as one of learnability of models by readers (e.g., by considering error / success rates, time required).

The CDNs framework is broadly formulated, being meant to be comprehensible by non-specialists, yet it captures significant results from psychology and human-computer interaction. It includes fourteen terms which describe aspects that are cognitively-relevant to a model (Table 5). The main idea is that each term is instantly recognizable and should feel familiar as soon as it is encountered.

Instead of describing all the details of the model that an external audience might not be familiar with, one could describe its profile (or characteristics) in terms of the cognitive dimensions.

Index	Guideline	Description
1	Abstraction	“Types and availability of abstraction mechanisms.”
2	Hidden dependencies	“Important links between entities are not visible.”
3	Premature commitment	“Constraints on the order of doing things.”
4	Secondary notation	“Extra information in means other than formal syntax.”
5	Viscosity	“Resistance to change.”
6	Visibility	“Ability to view components easily.”
7	Closeness of mapping	“Closeness of representation to domain.”
8	Consistency	“Similar semantics are expressed in similar syntactic forms.”
9	Diffuseness	“Verbosity of language.”
10	Error-proneness	“Notation invites mistakes.”
11	Hard mental operations	“High demand on cognitive resources.”
12	Progressive evaluation	“Work-on-date can be checked at any time.”
13	Provisionally	“Degree of commitment to actions or marks.”
14	Role-expressiveness	“The purpose of a component is readily inferred.”

Table 5: Cognitive Dimensions of Notations – source: (Green, Blandford, Church, Roast, & Clarke, 2006)

2.2.4 Physics of Notations

The physics of notations (PoN) is a framework for “evaluating, comparing, improving and designing visual notations for requirements engineering” (Moody D. L., 2010). It was applied to identify design flaws, evaluate and provide practical suggestions for improvement of some of the leading software engineering and requirements engineering notations, such as: i* (goal modeling), Archi-Mate (enterprise architecture), UML (software systems), Use Case Maps (user requirements), Business Decision Modeling (decision modeling), and BPMN (business processes).

According to (Moody D. , 2009), “current notations consist of little more than dreaming up symbols and voting on them” and therefore need to be improved given that the main purpose of a notation is to support the formalization of business or IT processes in terms of decisions, and the rules that make up those decisions. (Moody D. L., 2009) mentions numerous advantages for making visual notations more cognitively effective: increased visual processing (faster understanding), integrated reasoning (locational indexing), precision (monosemy), concision, memorability, and ease of processing by computers. He mentions the lack of principles for evaluating and designing visual notations, as designers rely on “instinct, imitation and tradition”.

Index	Principle	Description
1	Semiotic clarity	“There should be a 1:1 correspondence between semantic constructs and graphical symbols.”
2	Perceptual discriminability	“Different symbols should be clearly distinguishable from each other.”
3	Semantic transparency	“Use graphical symbols that suggest their meaning.”
4	Complexity management	“Include explicit mechanisms for dealing with complexity.”
5	Cognitive integration	“Include explicit mechanisms to integrate information from different diagrams.”
6	Visual expressiveness	“Use the full range and capacity of visual variables.”
7	Dual coding	“Use text to complement (not replace) graphics.”
8	Graphic economy	“The number of different graphical symbols should be cognitively manageable.”
9	Cognitive fit	“Use different visual dialects for different tasks and audiences.”

Table 6: Physics of Notations – source: (Moody D. L., 2010)

PoN provides nine principles of notation evaluation that act as guidelines for “designing cognitively effective visual notations optimized for human communication and problem solving” (Table 6). The principles are synthesized from theory of how visual notations communicate and are based on empirical evidence (evaluations with readers).

PoN has several advantages:

- Principles are specifically developed for designing software engineering notations, but are based on general principles of visual perception and cognition, so are applicable to any visual notation regardless of domain.
- Visual notations that satisfy the principles are more cognitively effective than those that do not, so PoN can be used to provide recommendations for visual notations.
- Principles support rigorous, symbol-by-symbol analysis.

Based on our experience (Popescu & Wegmann, 2014) of applying PoN to analyze the SEAM notation (Appendix 5), PoN also has several limitations:

- PoN focuses on the physical (perceptual) properties of notations rather than their logical (semantic) properties. It does not capture systemic aspects, such as the fact that the notation is dependent on the modeler’s viewpoint and his interpretation of the reality he observes.
- Some principles contain sub-principles while others do not, e.g., “cognitive integration” (P5) refers to: conceptual integration (summarization and visual momentum) and perceptual integration (signposting, orientation and navigation map), whereas “dual coding” (P7) refers only to the use of text to complement graphics.
- Some principles overlap, e.g., “perceptual discriminability” (P2), “semantic transparency” (P3), and “visual expressiveness” (P6). Attempting to apply one principle leads to the consideration of another. Or increasing the conformity for one principle leads to the decreasing conformity for another one, e.g., increased “visual expressiveness” (P6) leads to a decreased “graphical economy” (P8).
- All principles are of equal importance. They can be summed up to give a total score measuring how “cognitively effective” a visual notation is. However, some principles might be more important than others (e.g., for a specific notation). As such, some might emphasize more meaning than others (e.g., text is necessary in conjunction with colored paper).
- PoN does not consider the recursion of concepts at different abstraction levels (e.g., a system can be modeled as a whole or as a composite, an IT organization can be part of a business organization), which contradicts “semiotic clarity” (P1).
- PoN considers a symbol-level analysis of the notation, but not an analysis of the whole model created with the notation that might include the expression of the problem (rationale) and the steps leading to a solution (story).

In Figure 16 we show how PoN applies to an example that considers the instantiation of an object “vehicle” to two objects: “car” and “truck”. The “car” is further instantiated with a “red Ferrari”, a “red Honda”, or a “grey Mercedes”. The truck is further instantiated with a “red truck”. The first principle implies that each semantic construct (e.g., car) has a different graphical symbol: “car” and “red Ferrari” are two distinct graphical symbols. The second principle shows an increased visual distance between graphical elements: readers can clearly distinguish between the four elements in

the last column. The third principle implies that symbols suggest their meaning: detailed photos / icons of objects are used. The fourth principle demands for explicit complexity management mechanisms: these are shown with the hierarchy, colors and borders. The fifth principle demands for mechanisms to connect information from different diagrams: the “red truck” transports “cars”. The sixth principle demands for the use of the full range of visual variables: used mostly shape, color, value, orientation and texture, but not size. The seventh principle demands the use of text to complement graphics: all symbols have labels marked with text. The eighth principle demands for a cognitively manageable number of graphical symbols: there are seven symbols used in total. Finally, the ninth principle demands for different visual communication dialects: abstract (icons to the left) for experts and concrete (pictograms to the right) for novices.

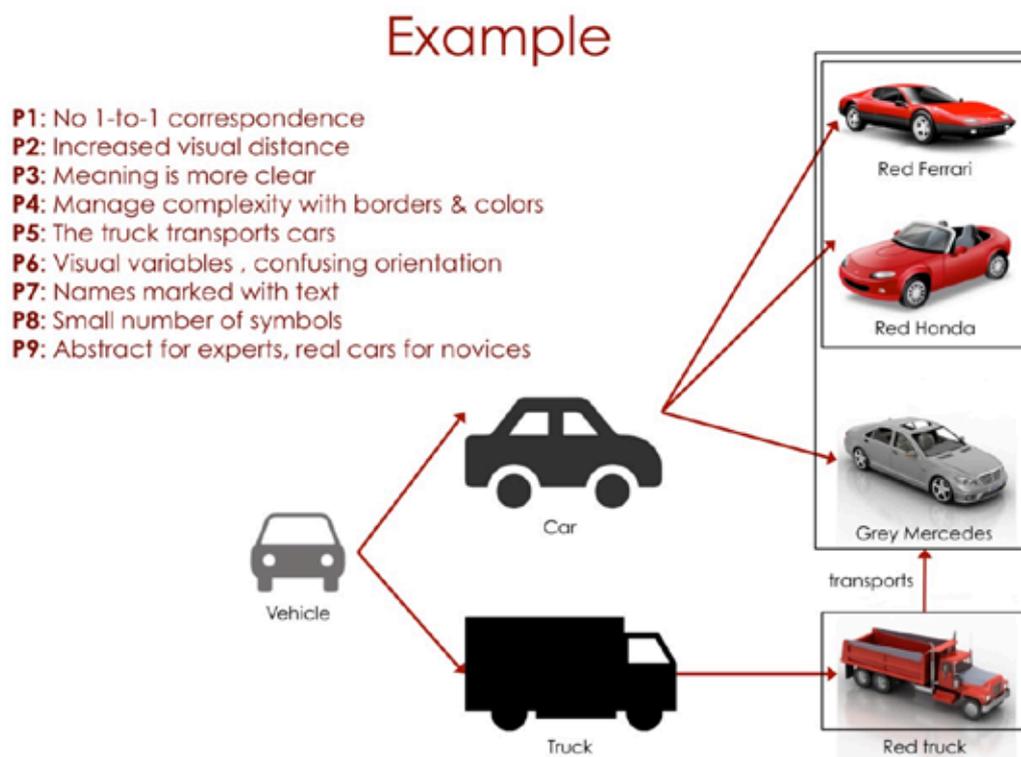


Figure 16: Physics of Notations principles applied to a vehicle example

2.2.5 Concluding Remarks

In Chapter 2.2, we have presented four notation-evaluation frameworks: SEQUAL, 7PMG, CDoN and PoN. They include guidelines for designers to evaluate and improve their notations. It is difficult to concretely apply these guidelines in practice, especially because they operate on different characteristics of the notation. Designers need to achieve a trade-off among guidelines.

Of all frameworks, PoN is the most widely used. We observe critical inconsistencies between PoN and hierarchical notations, such as SEAM. Given that in SEAM concepts are hierarchical, PoN proposes a different notation at each hierarchical level. However, this increases the model’s complexity as modelers need to use distinct graphical symbols for the entities they model. Therefore, it becomes challenging for readers to understand a model created with numerous graphical elements. In contrast, SEAM uses the same notation applied to all hierarchical levels (e.g., the shape for services / processes / actors is the same). PoN operates at the level of graphical elements within the model

rather than at level of model in its entirety. It does not include guidance for explaining the problem (rationale) and the steps leading to a solution (story).

2.3 Model Design Frameworks

We present two model design frameworks: design analysis and story phases. Design analysis is useful to understand both the modeler's rationale for creating the model and the actors' rationale for justifying the actions they make in models. Story-phases are useful to structure the modeler's story in the model so that readers can easily follow the main steps of the story presentation.

2.3.1 Design Analysis

Design space analysis (MacLean, Bellotti, & Shum, 1993) (DSA) is an approach for helping engineers reason about the design of the artifacts that they create. Visually, it is used to create a graphical representation which can help others understand the resulting design. In short, it is used to represent the design rationale (DR) by combining design information. The design rationale takes into account justifications for the design and possible alternative designs. Its aim is to provide modelers with a means to communicate their reasoning behind their models. A design rationale is a representation for documenting the argumentation of the modeler with respect to the graphical symbols that he uses to show entities in his models. The graphical symbols have to be understood by the model's readers. They are useful for reasoning, reviewing, managing, documenting and communicating. Different modeler's choices in the design space result in different interpretations of the readers. Modelers therefore need to analyze the design rationale to decide what to include in the model. Then, they can explore how to represent concepts graphically.

One method of design space analysis is the Toulmin method (Toulmin, 1958). It is an informal method of reasoning that involves data, claim and warrant of an argument. The data is the evidence used to prove something. The claim is what one is proving with the data. The warrant is the assumption that connects the data to the claim (MacLean, Bellotti, & Shum, 1993). The modeler's goal is to link the data with the claim by making the warrant or argument more explicit (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997). Differently put, the modeler needs to include evidence that readers can relate to in order to understand the warrant, and therefore the relation between the data and the claim, or between the model elements and the main question. The visual expressiveness of the graphical notation determines how the arguments are perceived by readers.

One expression of the design rationale is the Question, Options and Criteria (QOC) framework (MacLean, Young, Bellotti, & Moran, 1991). Questions are used to encapsulate key issues which focus on the creation of a model. They refer to "Which one?". Options provide possible answers to questions. Criteria are used to assess and compare options. Assessments form the relation between Options and Criteria and can be positive or negative (Figure 17 in green and red).

QOC is a semi-formal notation useful to identify key design aspects, visualize discussions, track unresolved issues, and quantitatively assess the strengths and weaknesses of different positions. QOC is similar to the Question, Idea and Argument developed by (Rittel H. , 2016). The relation between QOC and DSA is that DSA includes all possible Options for solving the Questions. Modelers do not need to consider the design space exhaustively, but make choices for what is relevant

for the situation at hand. DSA is therefore the process of finding key questions, exploring relevant options, and justifying the selection of options through criteria and their assessments for options.

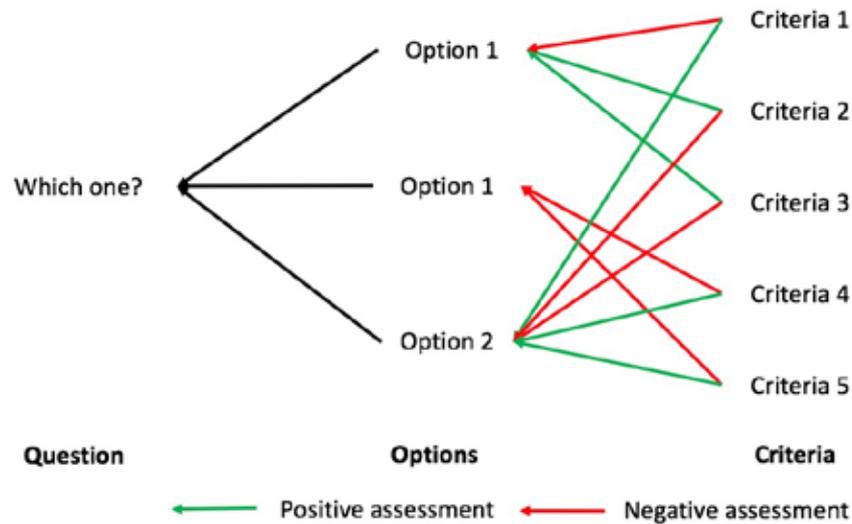


Figure 17: Question, Options, Criteria framework – adapted from (MacLean, Young, Bellotti, & Moran, 1991)

2.3.2 Story Phases

Modelers are interested to convey meaning in models they create. This meaning is easily interpreted by readers when they can understand what happens in the model, or the story that the modeler wants to convey. We present how the story can be structured (in story phases) to facilitate understanding.

There is no consensus about the number of story phases (model instances) in the scientific literature. Various authors propose to divide a story into:

- “4 stages”: what is the experience, when and where it takes place, and who and what are involved (context), high points developed through recapitulation of events (conflict), suspension of action at a crisis point (climax), resolution of the crisis by summarizing events (closure) (McCabe & Peterson, 1984)
- “4 essential phases”: setting and characters, challenge or problem with which the characters must cope, climax or turning point, resolution (Mortensen, 2009)
- “5 stages”: context, conflict, climax, closure, conclusion (Kautzer, 2012)
- “7 steps”: premise, scene ideas, characters, settings, outline, action (Klems, 2016)
- “7 steps”: characters, challenges, motivation, setting, obstacles, climax, closing (Storyjumper, 2010)
- “8-point arc”: stasis, trigger, quest, surprise, critical choice, climax, reversal, resolution (Hale, 2008)

In this research, we define a synthesis of the story-phases presented above. This synthesis includes the minimum number of story phases (four) that we found to be necessary for readers, by interviewing them, to understand stories. We combine the last two story phases proposed by (Kautzer, 2012) into one because we believe that closure and conclusion are essentially the same. We consider the first four phases proposed by (Kautzer, 2012) and explain each of them more precisely using the

definitions from (McCabe & Peterson, 1984). We apply these phases to create story-based models (detailed in the next chapters).

- *Context*: includes situation and actors
- *Conflict*: includes the problem and the rationale to address the problem expressed using questions, options, criteria and their assessments
- *Climax*: includes the solution to the problem
- *Closure*: includes the situation and actors, with no problem

These story phases enable modelers to develop a storyline or structure that is useful to unfold the story to readers.

One alternative to story-telling using chronological phases is the Who? What? Why? When? Where? How? (5W1H) framework (Jang & Woo, 2005). Originally developed and employed in journalism practices, the 5W1H framework helps categorize various aspects of the model, thus structuring the story, and facilitates information recall. 5W1H was applied to modelling value creation for e-commerces (Jang & Woo, 2005) and re-engineering for an identification service (Chung, Won, Baeg, & Park, 2009). 5W1H (with an extension including an extra “H” in 5W2H – “How long”) represents a taxonomy, or grouping, of story elements into categories that readers can easily understand and recall.

2.3.3 Concluding Remarks

In Chapter 2.3, we have presented two model design frameworks necessary for modelers to create models that show a story. The two are: design analysis and story phases. Design analysis helps modelers make the rationale of the actors more explicit in their models, enabling readers to understand the actors’ reasoning. The expression of design analysis is done by modeling the questions, options, criteria and their assessments. Story phases help modelers structure their story either chronologically or using a taxonomy. In this research we use the following four story phases: context, conflict, climax and closure.

2.4 Summary

In Chapter 2, we have presented three IT modeling notations, four notation-evaluation frameworks, and two model design frameworks. Most notations require readers’ expert knowledge (mainly IT). They do not consider different human perceptions of reality and assume an understanding of all concepts by all readers. They contain abstract shapes that are difficult to understand by non-IT readers. Furthermore, modelers of such notations seldom involve readers in the creation and evaluation of models.

Non-IT readers are confused by what the IT modelers communicate in their models. Models created by IT modelers lack the relation with reality, the rationale and the story. It is difficult for readers to understand (i) abstract graphical symbols without modeling the relation with the reality that they observe, (ii) the model actors’ reasoning for the choices they make without modeling the rationale, and (iii) what happens in the model that the modeler wants to convey without modeling the story. Furthermore, notations and models were previously only evaluated with readers for individual model instances instead of a sequence of model instances forming a story-based model.

In this thesis, we address these shortcomings. We create models by modeling the relation with reality, the rationale and the story, so that readers interpret a story that is as close as possible to the story that the modeler wants to show.

Chapter 3 Research Method

We present the method used in this research. The research is applied to creating models (Design Science) that are evaluated by people (Social Science). We give an overview of Design Science, the prevalent research method in information systems (IS). It emphasizes how information-systems artifacts (models) should be created. We explain why this research has industry and research relevance. It is based on the evaluation of models with people (Social Science) and helps modelers improve their models. We explain why this research has industry and research rigor. It complements existing notation-evaluation frameworks. This research is qualitative because we gain insights into how people understand models.

3.1 Model Creation and Notation Design

(Hevner, March, Park, & Ram, 2004) describe a framework for the design of information-systems artifacts named the Information-Systems research-framework. The design is the result of engineering, by working with information systems (which contain people, organizations and technology). It requires innovative ideas, practices, and objects that facilitate the efficient and effective analysis, design, implementation, and management of information systems. The framework emphasizes the creation and evaluation of artifacts intended to solve organizational problems: constructs (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices), and instantiations (implemented and prototype systems). The artifact's use, perceived usefulness, and effect on individuals and organizations, can be predicted and explained by theories.

We use the Information-Systems research-framework to design our artifacts. The artifacts presented in this research are our models. These models show information systems that include people, organizations and technology. We describe how these models were created iteratively with input from readers. We present our two innovations: (1) improvements of models and notations that use iterations that were evaluated by readers, and (2) recommendations for modelers and designers of modeling notations that were evaluated by modelers.

In Figure 18, relevance represents the usefulness in practice (i.e. for other practitioners) and is expressed through business needs that relate the environment with information-systems research. It leads to the practical question. Rigor represents the usefulness in research (i.e. for other researchers) and is expressed with applicable knowledge that relates the knowledge base with information-systems research. It leads to the research question. We add the practical question and the research question to the Information-Systems research-framework in order to make our contributions to both practice and research more explicit.

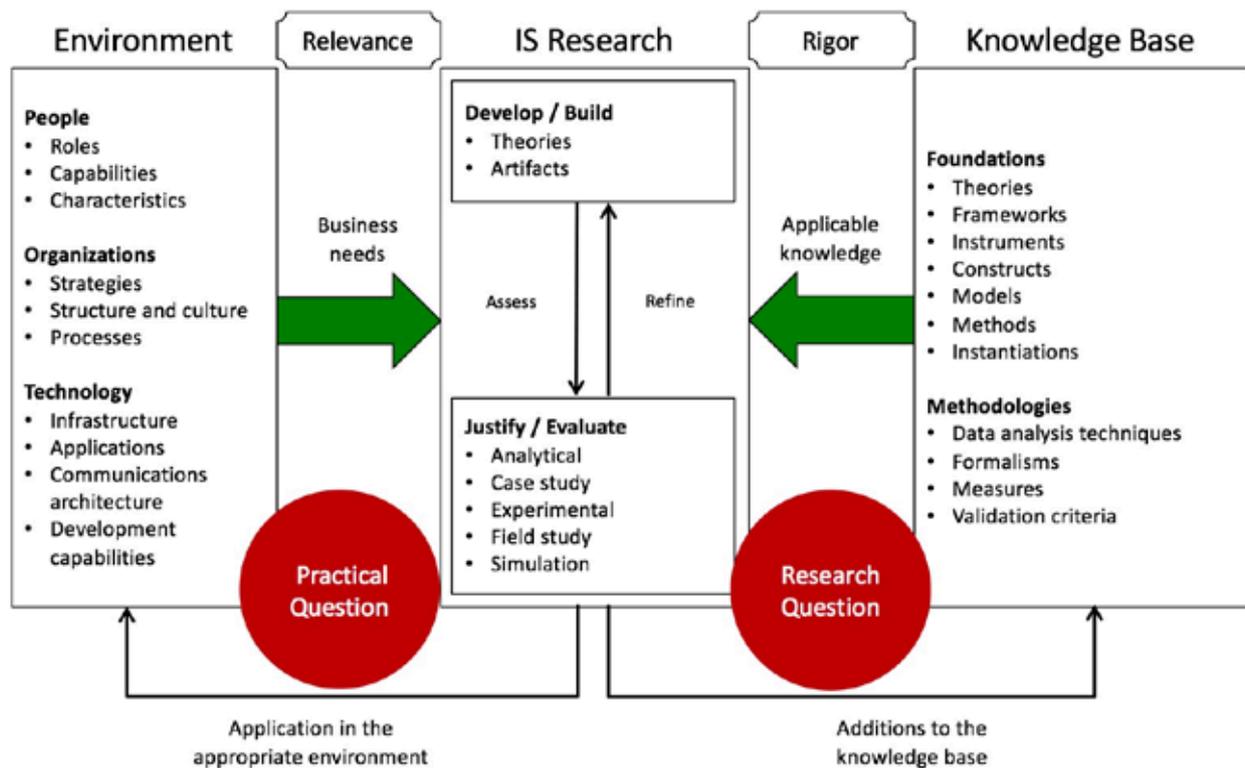


Figure 18: Information-Systems research-framework with the addition of the practical and research questions – source: (Hevner, March, Park, & Ram, 2004)

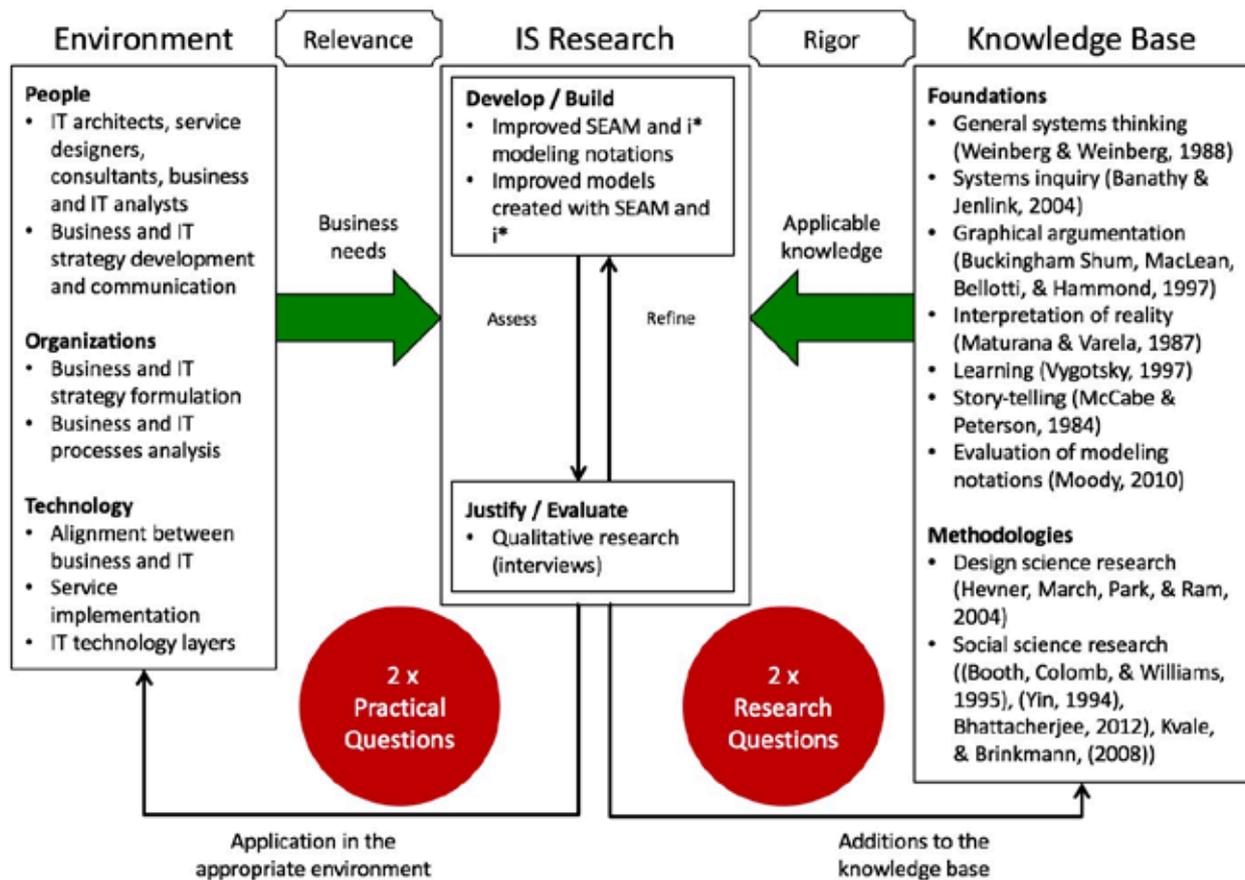


Figure 19: Information-Systems research-framework applied to our research with the addition of the two sets of practical and research questions

We instantiate the Information-Systems research-framework to our research (Figure 19). This instantiation can be done to consider either the model (storytelling using one model supported by the notation) or the notation (storytelling using all models supported by the notation).

In terms of the Design-Science paradigm, the environment, that this research addresses, is formed of people (IT architects, service designers, consultants, business and IT analysts) who work on business and IT strategy development and communication. Within their organizations, they formulate business and IT strategies and analyze business and IT processes (e.g., during workshops with non-IT people). The technology they work with concerns the alignment between business and IT, service implementation, and IT technology layers. The information systems research performed in this thesis concerns the development of improved modeling notations (e.g., SEAM and i*) in order to improve models created with these notations. Finally, the knowledge base of the current research consists in work related to the following streams of research: general systems thinking (Weinberg & Weinberg, 1988), systems inquiry (Banathy & Jenlink, 2004), graphical argumentation (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997), interpretation of reality (Maturana & Varela, 1987), learning (Vygotsky, 1997), story-telling (McCabe & Peterson, 1984), and evaluation of modeling notations (Moody D. L., 2010). The methodologies used in this research are related to Design-Science research (Hevner, March, Park, & Ram, 2004) and social science research (Booth, Colomb, & Williams, 1995), (Yin, 1994), (Bhattacharjee, 2012), (Kvale & Brinkmann, 2008)).

(Hevner, March, Park, & Ram, 2004) present seven guidelines for conducting, evaluating and presenting Design-Science research. Table 7 includes these guidelines, along with their description and a discussion of how the current research meets these guidelines.

Guideline	Description	Discussion
1. Design as an Artifact	“Design Science research must produce a viable artifact in the form of a construct (vocabulary and symbols), a model abstractions and representations), a method (algorithms and practices), or an instantiation (implemented and prototype systems).”	The main artifacts produced through this work are SEAM and i* models based on modeling the relation with reality, the rationale and the story more explicitly .
2. Problem Relevance	“The objective of Design-Science research is to develop technology-based solutions to important and relevant business problems.”	This work addresses the following questions: <ol style="list-style-type: none"> 1. How should modelers create models so that readers understand the story that the modeler wants to show? 2. How should designers design notations so that modelers create models that readers understand?
3. Design Evaluation	“The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well executed evaluation methods.”	The design evaluation method belongs to qualitative research. We conducted interviews with people and demonstrated initially misaligned, then aligned interpretations of the modeler’s story by model readers.
4. Research Contributions	“Effective Design-Science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.”	This work contributes to modelers’ comprehension of the model creation by providing them with guidelines for showing the implicit relation with reality, the rationale and the story more explicitly. It identifies, through interviews, the implicit elements in models created with two IT modelling notations, SEAM and i* . It contributes to evaluating models based on the underlying story .

5. Research Rigor	“Design-Science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.”	The proposal from the current research is based on: general systems thinking, systems inquiry, graphical argumentation, interpretation of reality, learning, story-telling, and evaluation of modeling notations.
6. Design as a Search Process	“The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.”	We develop our search for an effective artifact (model) by identifying model elements that can be made more explicit to help readers understand the story, without a legend, audio or video , while maintaining the identity of the modeling notation.
7. Communication of Research	“Design-Science research must be presented effectively both to technology-oriented, as well as management-oriented audiences.”	To facilitate the communication of the research results presented in this work to a wider community, we developed a set of recommendations for modelers and designers of IT modeling notations and a modeling and a design process to help them use these guidelines.

Table 7: Information-Systems research-framework applied to the current research – source: (Hevner, March, Park, & Ram, 2004)

3.2 Research Relevance

Design-Science teaches us how models should be designed. It does not mention how models should be evaluated. Social-Science research emphasizes the ways of evaluating models with people.

(Booth, Colomb, & Williams, 1995) describe a practical guide of how research should be planned and carried out in any field (“craft of research”). They describe the following steps:

1. Find an interest in a broad subject area;
2. Narrow the interest to a plausible topic;
3. Question that topic from several points of view;
4. Define a rationale for the project.

We apply the steps proposed by (Booth, Colomb, & Williams, 1995) to our research aimed at helping IT modelers. We say that:

1. Our interest lies in model creation;
2. Our topic is modeling two examples: car-service maintenance and meeting scheduler;
3. We ask the question: “How should *a modeler* create models so that readers understand the story that he wants to tell?”. We additionally ask: “How should a *SEAM / i* modeler* create a model so that readers understand the story that he wants to tell?”.
4. We help modelers create models that readers understand.

We apply the steps proposed by (Booth, Colomb, & Williams, 1995) to our research aimed at helping IT notation designers. We say that:

1. Our interest lies in notation design;
2. Our topic is designing two modeling notations: SEAM and i*;
3. We ask: “How should *a designer* of modeling notations design notations so that readers understand the story that the modeler wants to tell?”. We additionally ask: “How should a *SEAM / i* notation designer* design a notation so that readers understand the story that the modeler wants to tell?”.

4. We help designers design notations for modelers to create models that readers understand.

By applying the template leading from a question to its significance to this research, as suggested by (Booth, Colomb, & Williams, 1995), we state, “We study modeling notations, because we want to find out how designers of modeling notations should design notations that help modelers to create models that readers understand. This applies to SEAM / i* modelers”.

(Yin, 1994) mentions only one study-question that we formulate as a research question and a practical question by using the guidance provided by (Booth, Colomb, & Williams, 1995). The research question for modelers is: “How should a modeler create a model so that readers understand the story that he wants to tell?”. The practical question is: “How should a SEAM / i* modeler create a model so that readers understand the story that he wants to tell?”. The research question for designers is: “How should a designer design a notation so that modelers create models that readers understand?”. The practical question is: “How should a SEAM / i* notation designer design a notation so that modelers create models that readers understand?”. (Booth, Colomb, & Williams, 1995) provide guidance to answer both the research questions and the practical questions. The modeler uses the notation to create models. He cannot change it. The designer can change the notation.

As mentioned before, the challenge is to improve the model’s underlying notation while maintaining its identity. This is important because designers want to promote their notation to modelers and readers, and recognize this notation in the models that they and other modelers create with readers. This represents the relation with the practical question. If we would have addressed only the research question, then the result of this work would have been the design of new notations, ones that do not necessary maintain the identity of the original notation. We therefore propose an improved notation, one that maintains the identity of the original notation.

We study two IT modeling notations, SEAM and i*. In order to answer the practical and research questions, we perform Social-Science research. More specifically, we perform interviews to evaluate how readers interpret models. We show how models can be created for readers unfamiliar with the modeling notation. If the readers, using solely the model, without a legend or an audio / video / textual explanation from the modeler, interpret the modeler’s story, then we consider the model’s design successful, hence the model is improved.

(Yin, 1994) mentions that case-study research is one of several ways of performing Social-Science research, along with experiments, surveys and analysis of information. A case study is “an inquiry that investigates a contemporary phenomenon within its real-life context, especially when boundaries between phenomenon and context are not clearly evident” (Yin, 1994). We apply case-study research to answer our “how” questions, because we, as research investigators, have little control over people’s interpretation of models. This case study is exploratory, because we explore model and modeling improvements. We show how models can be created for readers unfamiliar with the modeling notation.

According to (Yin, 1994) there are five components of research designs (based on “design methods for case-study research”):

1. Study question: “How” and “why” questions for case studies.
2. Proposition: What should be examined within the scope of the study.

3. Unit of analysis: The (group of) people evaluating the proposition.
4. Logic linking the data to the propositions: The data matches one proposition better than other propositions.
5. Criteria for interpreting findings: Comparing rival propositions, based on a set of criteria.

The research design presented by (Yin, 1994) indicates what data needs to be collected (or evidence); this is called the structuring of the case study (first three components). (Yin, 1994) the indicates what needs to be done with the data; this is called data analysis (last two components).

We follow the structure proposed by (Yin, 1994) to define the five components for our research:

1. Study question: “How should a modeler create a model so that readers understand the story that he wants to tell?”
2. Propositions: Numerous model iterations, some of which are included in the Appendix 1 and 2, from the original models to the proposed ones.
3. Unit of analysis: Readers who evaluate model iterations and provide suggestions for improvement.
4. Logic linking the data to the propositions: Readers’ interpretation of the modeler’s story.
5. Criteria for interpreting findings: Improvements from one model iteration to the next, as perceived by readers, with respect to the modeler’s story. Model iterations need to maintain the identity of the notation.

(Hevner, March, Park, & Ram, 2004) teach how artifacts should be designed, in our case create models. (Yin, 1994) and (Booth, Colomb, & Williams, 1995) teach how models should be evaluated by people, in our case model readers. (Strauss & Corbin, 1998) teach how qualitative research should be performed in order to develop “grounded theory”, in our case propose recommendations to modelers and designers.

We create improved models that readers understand. We help modelers create improved models. We help designers design improved notations. The practical relevance is expressed with the practical questions. The research relevance is expressed with the research questions. Both practical and research relevance are based on the evaluation with readers of all model iterations that were created in this research.

3.3 Research Rigor

Design-Science considers rigorous research to rely upon the application of methods in both the construction and evaluation of the design artifact. Rigor consists in the applicable knowledge that relates the knowledge base with information-systems research and is generated from this research. It is expressed with the two research questions addressed to modelers and designers.

For this research, the knowledge base is represented by the current theories for evaluating and designing notations (notation-evaluation frameworks) applied to information-systems research. These theories focus on improving the visual syntax of notations, without considering the story that the modeler wants to convey (i.e., the model’s meaning). We formulate contributions for modelers and designers of modeling notations.

For modelers to create models that readers understand, we recommend modeling (i) the relation with reality, (ii) the rationale and (iii) the story. These recommendations apply to the level of model creation. We propose a modeling process that shows the main steps that modelers need to follow to create improved models.

Similarly, for designers to design notations that readers understand, we recommend designing (i) the relation with reality, (ii) the rationale and (iii) the story. These recommendations apply to the level of notation design. We propose a design process that shows the main steps that designers need to follow to create improved notations.

3.4 Research Process

The following research process (Figure 20) was applied to improve two models developed with two modeling notations, SEAM and i^* (Chapter 4 and Chapter 5, respectively). It enabled us to perform a qualitative research in order to develop “grounded theory” on designing notations and creating models that readers understand (Strauss & Corbin, 1998). It can be applied to improving other notations as well.

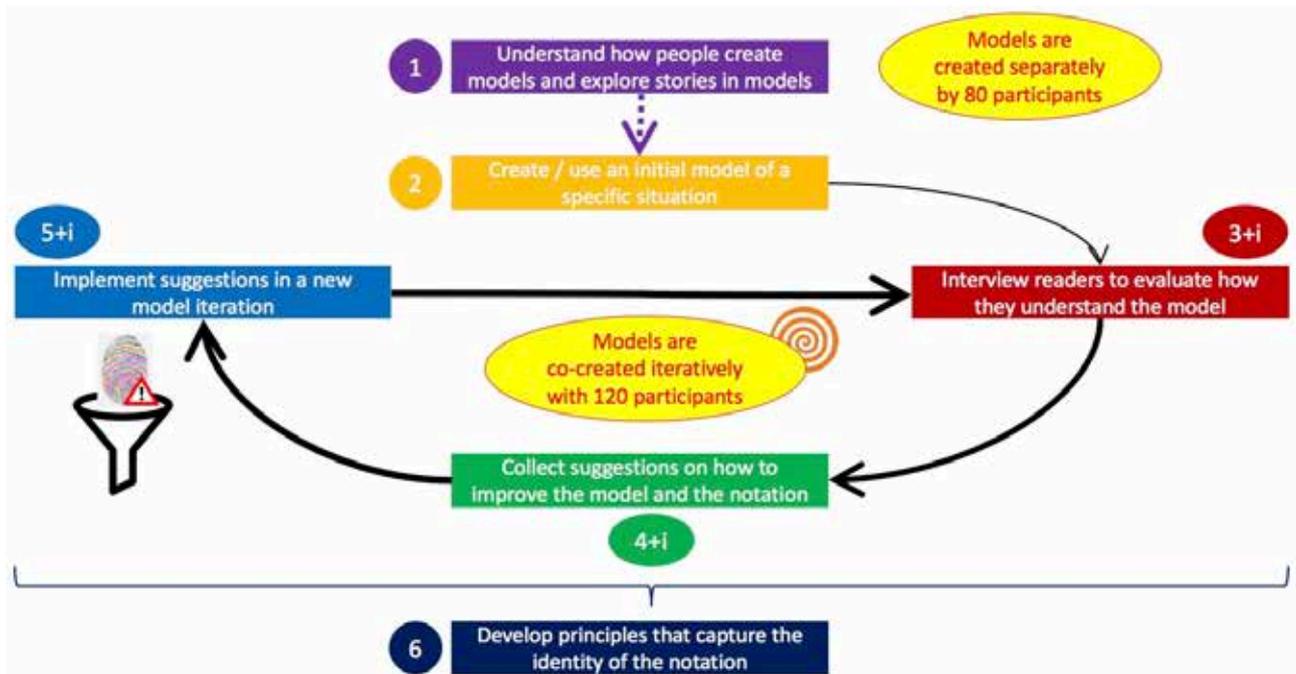


Figure 20: Research process

Steps 1 to 6 represent the main phases of the research performed in this thesis. Variable i used for steps 3, 4 and 5 shows the number of iterations (or full cycles).

In Step 1, we carried out a preliminary study in which we asked seventy-six people to draw one service model and one hierarchical (org chart) model of the organization in which they worked. Study participants were not offered any indication about what such models could entail. One hundred and fifty-two models were analyzed. More details about this study are included in Appendix 7.

Participants did not encounter problems when drawing hierarchical models of their organizations; they used tree-like structures such as those included on corporations’ websites. However, they did not know how to model a service organization. Each participant had a different understanding of the

concept of service and, consequently, drew it differently (e.g., variations of flow-chart, ontology, and tree-like structures) than other participants. Examples of such models are shown in Figure 21.

The major outcomes of this preliminary research were the oral descriptions offered by the participants explaining their models. Numerous details were implicit and known to the interviewees, but were not shown in the model. All these models showed the story of “who does what and for whom”, but it was difficult for other readers, by looking solely the model, to understand this story.

Interviewees used plain language to describe their role in the organization, their major tasks, and how people work together to attain objectives. None used specific terminology. Participants often used the same graphical symbols in their models to represent different entities that they observe in their organization. They also used different graphical symbols for the same entities. Some participants used a large number of entities in their models, whereas others used a large number of relations among relatively few entities. It is therefore challenging to create standardized service models that numerous people can understand.

This study enabled us understand how people create models, gather insights about how they perceive reality, and explore new ways of creating models using stories.



Figure 21: Service models drawn by people using an ad-hoc notation

Steps 2-3-4-5 were done in a separate main study with one hundred and twenty participants, unfamiliar with the notations used to create models. These readers evaluated our models and provided feedback for improvement.

In step 2, we created (for SEAM) and used (for i*) an initial model of a specific situation. Then, for steps 3-4-5, models were co-created iteratively with readers, until the readers correctly interpreted the modeler’s story.

In step 3, we interviewed readers to test if they understand the current model iteration. Without providing readers any information about the model, we asked them: “What do you think the model is about / it suggests?” and recorded the audio of the readers explaining the model to the interviewer, who took notes. In the audio readers express, sometimes with hesitation, their search for meaning, i.e. what they think that they understand from the model.

In step 4, we collected readers’ suggestions on how to improve the model. We used the audio transcriptions and the interviewer’s notes to summarize readers’ feedback. The suggestions included what elements to include or not and how to model them (e.g., with what shapes, colors, positions, as well as their relations).

In step 5, we implemented possible model improvements (either ours or proposed by readers who evaluated the models) that maintained the identity of the modeling notation, in a new model iteration. The maintaining of the modeling notation represents our filter for creating improved model iterations. We considered all readers' suggestions, but proceeded to implement only those that are in accordance with the notation's underlying principles (Chapter 2.1.1 and Chapter 2.1.2) by verifying our improvements with expert modelers (e.g., for SEAM we asked Professor Alain Wegmann and Dr. Gil Regev, for i^* we compared with other models and the description of the i^* modeling method). For instance, we did not create any SEAM model iteration in which the context is not shown because this represents a fundamental SEAM principle. We do not proceed to the discovery of new notations, but rather to the constrained discovery of modified notations. We look for ways to make the notation more understandable, without changing its identity.

The complete iteration cycle (steps 3-4-5) builds on one model iteration together with the participants' suggestions (input) to produce an improved model iteration (output). The newly obtained model iteration (at the completion of one cycle showed using the "+i" in Figure 20) was re-evaluated with other readers. This evaluation enabled the modeler to create an improved model iteration ("i+1") by applying the steps 3-4-5. Improvements are marked relative to the previous model iteration. The readers who evaluated the last iteration correctly interpreted the modeler's story. They did not evaluate any previous iterations.

Steps 3-4-5 were also applied in a Bachelor project by a student who worked under the supervision of the thesis author. The project consisted in exploring creativity techniques (e.g. 3D paper, displacements of model elements triggered by model readers) to improve models (Soccard, 2016). The project was performed over a period of four months with the student working 20 hours per week on the project. The author of the thesis and two SEAM modelers conducted review meetings every two weeks. During these meetings new ways of showing model elements were presented. Some were beyond the suggestions of interviewees.

In step 6, we started identifying the principles underlying the notation. These principles capture the identity of the notation. We explored these principles for SEAM mostly. Future work will consider an in-depth understanding of these principles.

Participants' feedback was manifested in reactions, explanations, questions, and suggested improvements. The evidence collected relies on documents (investigator's notes), audio recordings, and direct observation. We identified the key triggers for the readers' confusion because of the ambiguity of the notation. For instance, readers noticed that in some cases, some graphical elements were not present in the model (e.g., the modeler's goal, the problem that the model depicts). We added them. Some readers had problems interpreting various graphical elements either individually or in relation with others. We considered graphical elements that are more easily interpretable and used them in our models.

The evaluation metric used by the investigator to evaluate the participants' comprehension of the model is the identification of the story. The interview participants were asked to tell, based solely on the information included in the model, what they thought the model is about. If they would interpret an element of the model differently than what the modeler wanted, then this element would be updated with one that helps the readers to more closely interpret what the modeler wanted to express, in the following model iteration. The constraints considered were that the model should not

be in textual, audio or video format, but only graphical and printed on paper, without a legend or other explanations (these constraints consider the practical problem of the modeler). The last model iteration represents the proposed model, the best that the modeler (in his role as research investigator) produced, given the feedback from readers.

The investigator determined the issues that the participants identified and grouped them into three categories:

1. *Relation with reality*: what do the elements (e.g., photos, icons, terminology) represent?
2. *Rationale*: what solution is chosen based on the problem (e.g., why is it meaningful)?
3. *Story*: what happens (e.g., series of occurrences)?

In creating the model, the modeler needed to maintain the identity of the modeling notation expressed with the modeling principles specific to the notation. He decided upon the degree of abstraction or concreteness to be shown in the model so that participants would understand his story. A fully abstract model contains the principles of the notation and is useful for expert modelers. A fully concrete model instantiates the abstract model to one specific case by emphasizing the relation with reality and is most useful for novices. Therefore, the modeler needed to choose how to create the model based on the continuum between abstraction and concreteness. From our research process, we observed that expert modelers (familiar with the notations) seek to understand the essence of the notation whereas novices (unfamiliar with the notation) seek to understand the example presented to them in the model, rather than the principles that govern the notation.

Some of the suggestions proposed by the interview participants could not be incorporated in the model due to the identity of the notation. Some participants' suggestions contradicted those of other participants (e.g., use of color, size of elements, position of elements). Other suggestions were difficult to implement (e.g., model all possible changes in the actors' states). We checked the implementation of readers' suggestions against the identity of the modeling notation. We asked modeling experts to verify that the proposed iterations are in accordance with the principles of the (SEAM / i*) notation. This was to ensure that a different notation was not produced.

3.5 Interviews

We present in detail the organization of the interviews for the main study. We used the seven stages of an interview investigation (Kvale & Brinkmann, 2008) in order to organize our investigation. The seven stages are: thematizing, designing, interviewing, transcribing, analyzing, verifying and reporting. The below synthesis of the interviews concentrates on the last five. We explain what happened both during and after the meeting with interviewees.

We interviewed one hundred and twenty readers in total during 2014-2016. One hundred readers evaluated twenty-three model iterations created with the SEAM notation. In total, twenty readers evaluated twelve model iterations created with the i* notation. For each notation, we modeled a specific story example.

Interviews were structured as informal discussions (Figure 22). They lasted between 30 and 60 minutes per participant. They took place at the Ecole Polytechnique Fédérale de Lausanne, at the interviewees' homes, their workplaces, or in public places. Participants were (undergraduate and

PhD) students, secretaries, analysts, managers, doctors, lawyers and economists. All participants were readers unfamiliar with the notation. Interviewees were presented with the model printed on several pages, each page representing a model instance. They were asked to explain the model fully (“What do you think this model is about?” / “How can you explain this model?” / “What do you think is the story behind this model?”), without being offered any details about the story (content) or the notation. Participants were audio-recorded while explaining to the interviewer the story that they perceived. The interviewer wrote remarks that he found interesting (Figure 22). When observing the model instances, participants often realized some details or hints that enabled them to arrive at “Aha!” moments and re-frame the story differently. Based on participants’ feedback, the modeler (thesis author and research investigator) created new iterations so that other readers would understand the story better. He re-tested these new improvements with other interviewees.



Figure 22: Interview with one participant interpreting a model

When interpreting one model instance at a time, readers did not understand how and why some elements were linked. With the interpretation of a (partial) sequence of model instances, they were able to relate to the information presented in the previous model instances, hence develop a story that made sense to them.

3.6 Summary

Design-Science provides a general framework for artifact-based research. It insists on demonstrating the usefulness of the artifact, but does not provide specific guidelines on how to go about this evaluation. Social-Science concentrates on the evaluation of models with people, but does not provide guidelines on how to design artifacts (models). We therefore combine both Design-Science and Social-Science research. We have presented the research method that was applied to create and evaluate models using two notations: SEAM and i^* .

In the following two chapters, we present the application of the research method to modeling with SEAM and i^* . A summary of our case-study research is presented in Table 8. For both notations, we modeled examples inspired by experience (SEAM) and by research (i^*).

For SEAM, we used the example of a car-maintenance service. The model represents a simplified fictive case, inspired by reality. In Chapter 4, we include the original SEAM model (first iteration) and our proposed SEAM model (last iteration). Other model iterations are included in Appendix 1.

For i^* , we used the example of a meeting scheduler, as presented in (Yu E. S., 1997). In Chapter 5, we include the original i^* model (first iteration) and our proposed i^* model (last iteration). Other model iterations are included in Appendix 2.

Modeling notation	Example	Readers
SEAM	Car-maintenance service	100
i^*	Meeting scheduler	20

Table 8: Case study applied to two modeling notations: SEAM and i^*

These examples are commonly encountered in situations from daily life. They are simple and easy to understand by readers who are unfamiliar with the modeling notation. Readers do not need to have a car, know how the service of a car is done, know any particular meeting scheduler, or know how to schedule a meeting. Our goal using story-based models is to present to readers all elements that are necessary for them to understand the modeler's story.

Chapter 4 Application to SEAM

We show how we applied the research method to improve the SEAM notation, using a personal example of the thesis author. We model the relation with reality, the rationale, and the story. In brief, the story is the following: “How do George and Monica have their car serviced?” The SEAM modeler asks himself: “How do I, as a SEAM modeler, create a SEAM model so that readers understand the story that I want to convey?”. We propose improvements for the SEAM model. These improvements have been validated by SEAM experts and used in workshops (Appendix 4).

4.1 Research Protocol

We instantiate the steps of the research design from Chapter 3.4 as follows:

1. Create an initial model in which two customers service their car
2. Interview readers to evaluate how they understand the story that the modeler wants to tell
3. Collect suggestions on how to improve the model and the notation
4. Implement suggestions in a new model iteration (while maintaining the identity of the SEAM notation)

It was necessary to interview one hundred readers and create twenty-three model iterations by repeating steps 2-3-4 until the modeler’s story was correctly interpreted by readers. Therefore, on average, between four and five readers evaluated one model iteration. With the last model iteration, readers could interpret the modeler’s story correctly. These readers did not evaluate any previous iteration. Some iterations of the SEAM model, the ones that include the most meaningful changes compared to the previous iterations, are included in Appendix 1.

4.2 Research Design

The five components of the research design (Chapter 3.2) applied to SEAM are:

- Study question: “How should a SEAM modeler create a model so that readers understand the story that he wants to tell?”
- Propositions: SEAM model iterations from the original model to the last model iteration presented in Chapter 4.5.3.
- Unit of analysis: Readers who evaluated and gave feedback on SEAM model iterations.
- Logic linking the data to the propositions: Readers’ interpretations of the modeler’s story.
- Criteria for interpreting findings: Improvements from one model iteration to the next, as perceived by readers, with respect to the modeler’s story. According to the notation designer, model iterations need to maintain the identity of the notation.

4.3 Interviews

We use the example of the following story: “How do George and Monica have their car serviced?”. We are interested in creating a SEAM model, so that readers understand this story.

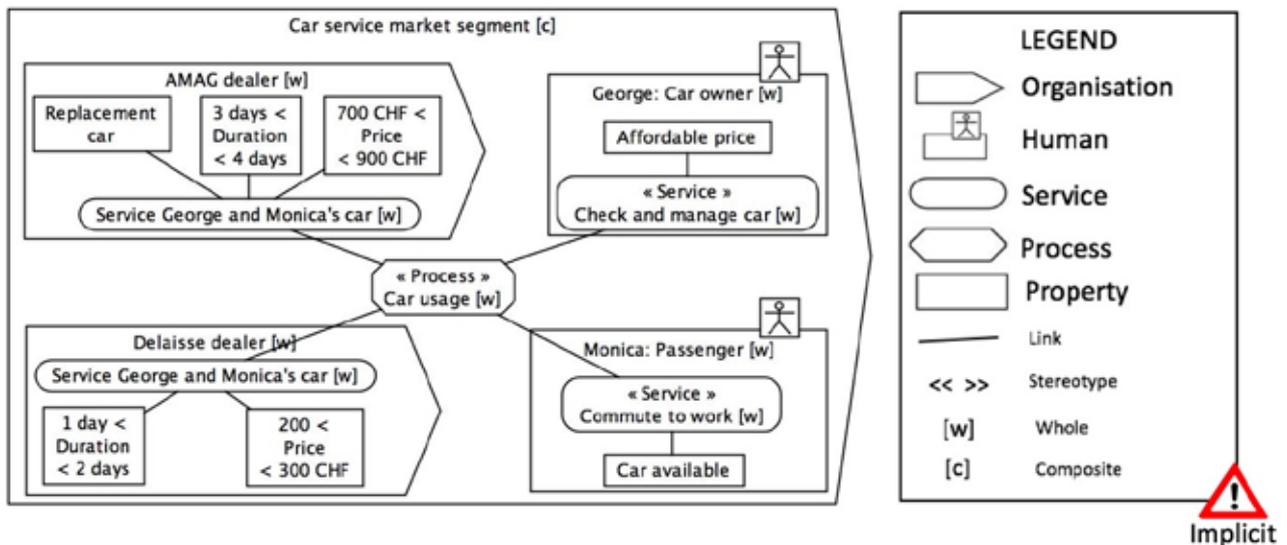


Figure 23: SEAM initial model

The initial model is shown in Figure 23. According to a SEAM modeler, this model includes one hierarchical level: a market segment (shown as a composite) with four components: actors shown as wholes (two suppliers (AMAG and Delaisse) and two customers (George and Monica)). The whole view enables the modeler to show the services provided by each actor and their relationship via the process shown in the middle. Each supplier offers a service to the customers. George and Monica need to choose one of the two dealers to service their car. The two customers form a family and each of them considers certain criteria. These criteria are reflected in the service offerings of the two dealers. Comparing the criteria with the service offerings, the customers choose one of the two dealers. The car is serviced by the chosen dealer and returned to the customers.

Further details of the story can be foreseen, such as the fact that Monica and George initially looked at the available dealers near their place of residence. Based on their criteria, for instance time and money, they choose one dealer. The dealer does the service, then the customers pay, and the car is returned to them.

We observed that readers need the text and the legend to understand the model. They also need the definitions of wholes and composites, services, processes, properties and relations. But this is not enough. It is difficult for modelers to correctly and fully write the textual description of the model. It is also challenging to create an audio or video narration to explain all the intricacies of the model.

4.4 Feedback

Understanding the SEAM model requires knowledge about the SEAM notation. We highlight the elements that are difficult to comprehend (that we call implicit) by readers. We group them into three sections: relation with reality, rationale and story. We include the questions that participants found useful to know (i.e. that are related to the example but not to the model shown).

4.4.1 Limited Relation with Reality

Interviewees were confused by concepts used in SEAM, such as system, service or process. They had difficulties understanding the relation between the customer and the supplier sides, in terms of service offering and the implementation of the service.

The SEAM notation shows wholes inside composites. Wholes, or black-boxes, are non-transparent systems (observers cannot perceive the inside). For the modeler, the external observer perceives only the holistic system with its emergent behavior (or service). Composites, or white boxes, are transparent systems (observers can perceive the inside). The decomposition of transparent objects (composites) with non-transparent objects (wholes) is applied in SEAM for both the construction or structure of systems, as well as for their behavior or action.

Readers are not accustomed to these conventions. The notion of whole and composite is unclear. Readers perceive the “w” and “c” signs but do not know how to interpret them. They might think that a whole is an object in its entire, complete form. “w” is used for a whole, or a black-box, but the “white box” begins with the same letter (“w”). Furthermore, a black-box is associated with a perceived behavior but not with an internal structure (wholes inside the composite). Some readers find it relatively easy to perceive an object made out of other objects, but they find it difficult to perceive an unordered decomposition of an action into sub-actions.

Each specific shape has a certain meaning for the SEAM modeler:

- *Rectangles*: properties (inspired by UML (Larman, 2005))
- *Ovals*: services (inspired by UML (Larman, 2005))
- *Hexagons*: processes (inspired by UML (Larman, 2005))
- *Arrow-box*: actors (inspired by the value chain of (Porter, 1980))

Certain conventions are used. Properties are linked to services. Services are linked to processes. Properties and services are shown in system as wholes. Processes and sub-systems are shown in systems as composites. All these conventions are not comprehensible by readers. For instance, they do not remark that services are linked to processes; they interpret that systems are linked together.

In model iterations, we evaluate numerous options for improving the model’s relation with reality:

- Including a secretary and a salesman for the two dealers;
- Considering George and Monica separately or together (as a family);
- Including the car to offer a service (“transportation”);
- Including another service provider (sub-system) in the dealer’s composition;
- Including a receptionist (single-point-of-contact) for each of the dealers;
- Showing the modeler’s viewpoint;
- Using circles, pluses and minuses for neutral, positive and negative criteria, respectively;
- Including a picture of a replacement car;
- Deciding which colors to choose (red for problem or for Monica);
- Showing the price constraint with an empty wallet metaphor;
- Showing the time constraint with a clock metaphor.

The following questions were asked by participants regarding the relation with reality. We provide the answers to these questions as well.

- Whose viewpoint is shown? – George’s viewpoint
- What is the relation between George and Monica? – They are a family
- Is the price fixed or can it be increased or decreased based on the service provided? – The price might fluctuate
- Who is responsible for the car? – The dealer during the entire service and the customers before and after the service
- What happens if the maintenance service goes wrong? – The dealer is responsible
- Who is the contact person from the two dealers? – The receptionist

In Chapter 4.5.1, we show explorations to answer these questions in our model iterations.

4.4.2 No Rationale and No Choice

The initial model does not explain the rationale leading to the actors’ choice of one dealer over the other. Readers were confused as they could not identify this rationale.

The following questions were asked by participants regarding the rationale. We provide the answers to these questions as well.

- Why do George and Monica need to service the car? – Regular service at 30’000 km
- Who chooses the dealer? – George and Monica together
- Why is the car important to Monica? – Monica needs it to commute to work every work day
- Is the service qualitatively different between the two dealers? – It might be
- Are Monica and George’s preferences strict or flexible? – Monica is flexible with money but not with time; George is flexible with time but not with money
- Is there a conflict between what Monica and George want? – This depends on the dealer’s value proposition
- What will George and Monica decide if AMAG performs the service faster but at a higher price? – This is not important as it is out of context
- What happens if no offer from the dealers corresponds completely to Monica and George’s needs? – They will still have to make a choice, one that does not fully satisfy their criteria

In Chapter 4.5.2, we show explorations to answer these questions in our model iterations.

4.4.3 No Story

The single instance of the model did not enable readers to understand how to interpret the story because it represents only one moment in time. Readers were confused as they could not identify a sequence of instances.

The following questions were asked by participants regarding the story. We provide the answers to these questions as well.

- Whose car is it? – It is George and Monica’s family car but George owns it
- Who takes the car to the garage for the service? – George

- Who pays for the service? – Both George and Monica in half
- Can the car be used temporarily while it is being serviced? – No
- How is the car serviced? – This is not important as it is out of context

In Chapter 4.4.3, we show explorations to answer these questions in our model iterations.

The readers' initial reactions of the model from Figure 23 show little comprehension of the modeler's story. They were: "I'm not sure what I'm looking at...", "I don't know what the model is showing!", "This thing with the service is confusing to me...", "To me this is the same thing.", "Off... I don't know, I'm sorry, my brain is not working". As the model improved, we noticed reactions that show a better comprehension. For instance, readers asked questions about the details of the graphical symbols rather than their meaning (e.g., "Why if the picture of the inside of the dealer used, instead of the same one as before?"; "What do the 35 minutes represent – commuting time by public transport or commuting with another car?").

4.5 Contributions to SEAM modelers

We present the contributions to SEAM modelers. These contributions fall under three main areas:

- *Relation with reality* by showing more explicitly the modeler's goal and model information, actors (human, objects and organizations), and the service offering (by using the readers' conceptualization);
- *Rationale* by showing more explicitly the main question, the options, the criteria, and the assessment of the criteria;
- *Story* by showing model instances corresponding to each story phase: context, conflict, climax and closure.

For each contribution, we highlight the improvements taken from the initial and final model iterations, as well as from intermediate model iterations presented in Appendix 1. The evaluation of each model iteration by the participants allowed the modeler to improve the way he showed the relation with reality, the rationale and the story.

4.5.1 Relation with Reality

We show how to make more explicit the relation between the model and the reality that the modeler observes, by (i) showing the modeler's goal and model information, (ii) modeling actors (humans, objects, organizations), and (iii) modeling the service offering, thus corresponding to readers' conceptualization of the situation instead of the modeler's conceptualization.

Modeler's goal and model information

The modeler's goal is made explicit by including it at the top of each model instance. SEAM models do not include the modeler's goal. Hence, SEAM modelers need to explain it orally when presenting models to audiences. By including the modeler's goal (emphasized in bold) on each model instance (Figure 24) readers can see the main idea (summary) of the story. Each model instance also includes other model information: the date when it was created, the name of the modeler, and his affiliation.

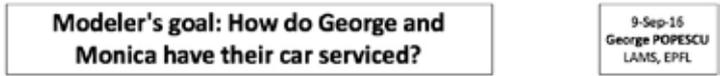


Figure 24: Modeler's goal and model information with the proposed SEAM

In our model iterations we explored if readers would find useful to include the modeler's goal explicitly in the model. The first three icons from Figure 25 created more confusion because readers did not distinguish between the various roles of George (actor in the model, modeler, interviewer, designer). The last icons were meant to replace the title of the model and explain visually the summary of the story: the car needs to be serviced, it is taken to a dealer, and then it is serviced. Readers preferred the textual explanation in the title of the model: "How do George and Monica have their car serviced?" together with textual information about the model's author rather than viewpoints for each system shown in the model.



Figure 25: Modeler's viewpoints and story summary from model iterations

Actors

There are three categories of actors used in the proposed SEAM model: humans (e.g., George and Monica), objects (e.g., car) and organizations (e.g., AMAG and Delaisse dealers). These are shown in Figure 26 with the current SEAM notation and in Figure 27 with the proposed SEAM notation (from the last model iteration). We explored different pictures of the actors. Some roles (e.g., George – husband and Monica – wife; AMAG / Delaisse – Audi dealer; the car – Audi car) were removed because they emerged from the model itself. Others were kept (e.g., Delaisse dealer, AMAG dealer) because some readers were not familiar with them.

We observed that the pictures need to be appropriate for the context that is shown. For instance, George is modeled as a student, emphasizing the low budget constraints, whereas Monica is modeled as a business lady, emphasizing her need to arrive in time for corporate meetings. George and Monica have separate individual goals (thus, we use two distinct photos). However, they need to decide (and are thus modeled) together, as a family (we use the photo of their flat entrance door) because they share a main goal. The three states of the car, (i) functional, (ii) needed to be serviced, and (iii) during service, are represented with (i) the hood closed, (ii) the hood closed and the indicator "Service due!", and (iii) the hood open, respectively (Figure 27). The two dealers are represented with a real photo, as perceived by the customers. We apply a 50% transparency to make the service offering more visible when overlaying other elements marked with photos, icons and text.



Figure 26: Human, object and organization actors with the current SEAM

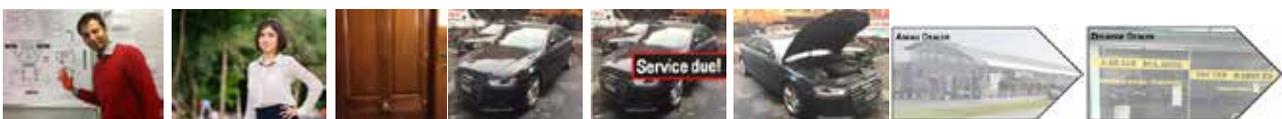


Figure 27: Human, object and organization actors with the proposed SEAM

We tested several pictures of the actors, the car, the two dealers, and other actors (Figure 28). First, we represented both George and Monica with a business photo; however, participants were confused about George’s goal of finding an inexpensive dealer when his material condition showed otherwise. We used the house metaphor in conjunction with George and Monica’s photo together, but readers appreciated as confusing to see twice both George and Monica. We also tested the photo of a house, wanting to show George and Monica as a family, but readers were lost in details about the house (place, size). Second, for the car, we tested with the SEAM notation showing the service, the metaphor of a car, the sketch of a car, and a generic car indicator. Readers preferred concreteness and wanted to see a photo of the real car and the real indicator. Third, to represent the two dealers, we tested with: the SEAM notation using the “+” sign, meaning that the organization includes the Local Audi Dealer and its partners, a generic dealer notation when the dealers were not known to George and Monica, the photo of an AMAG show-room taken from the Internet, a photo of the inside of a generic dealer, and a photo of the inside of the Delaisse dealer. Readers demanded to see in the model the elements as they are in reality. The difference between the outside (before the choice was made) and the inside (after the choice was made, when the car is in service) views of the dealer created confusion as readers thought of another dealer. Fourth, we explored including in the model the secretary for the dealer, a mechanic and a salesman. Readers considered these elements as details with respect to the service of the car.

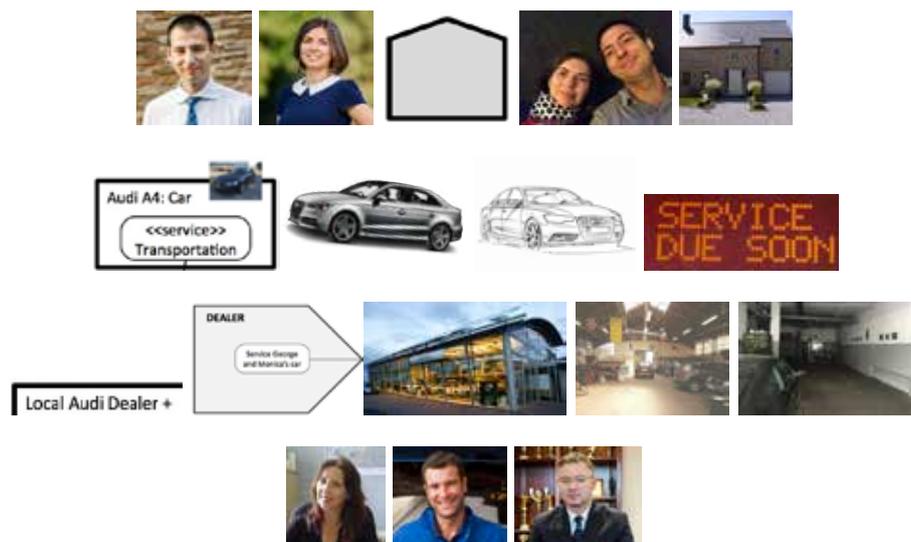


Figure 28: Human, object and organization actors from model iterations

Other graphical metaphors were used in relation to the actors (Figure 29). In order to represent the commute time (for Monica) with and without the car, we used the first and second icons overlaid on the photo of the car and the tram, respectively. We did not consider a taxi or a car rental because it is not important for the context shown. In order to represent money, we used the metaphor of a bag of money with a text suggesting George and Monica’s car service budget (1’000 CHF). The real banknotes (totaling 250 CHF) indicate the sum that was paid for the service and shows the embodied experience of the customers (especially George). We use icons when the sum of money is not precisely specified (budget) and photos when the sum of money is precisely specified (the sum of money of exactly 250 CHF was paid). We used a photo of the offer sent by the Delaisse dealer and a photo of the bill that was paid. The offer corresponds to an estimation, whereas the bill corresponds to a precise value that needs to be paid.



Figure 29: Graphical metaphors with the proposed SEAM

For money and time, we model the different granularities or units of measurement that are considered: the customers' goals ("lowest price" and "fastest service") and the dealers' offers (range of durations and prices) are presented vaguely using icons, whereas the budget (1'000 CHF), the commuting time (15 / 35 minutes), and the price paid to the dealer (250 CHF shown with real banknotes) are precisely defined. For the replacement car we used a photo of a real car (proposed by AMAG) instead of an icon in order to show the physical option that the two customers would get from one of the dealers.

In our SEAM model iterations (Appendix 1), we explored other graphical symbols (Figure 30). We wanted to show simply "time" using the first and second icon, but readers interpreted its significance too precisely: "one hour and fifteen minutes" or "9 hours and 10 minutes" – this was not important as the precise interpretation was out of context. We wanted to show simply "money" using the third icon but readers interpreted its significance too precisely: "20 CHF" – again out of context. We wanted to show that one dealer was not chosen and the other was chosen by using the thumbs-down / thumbs-up pair of icons but readers preferred only red / green color. We modeled the fact that George is sensible to money by using the icon of an empty wallet (to show the emotional pain related to paying for the service) and that Monica is sensible to time by using the icon of a young lady with an umbrella in the rain (to show the emotional pain related to not having the car available). We also explored showing what happens with George and Monica during the service of the car, by using photos of George's bikes and a photo of a bus in Geneva.



Figure 30: Graphical metaphors from model iterations

Service offering

The service offering refers to the characteristics of the service that the two dealers provide. It is shown in Figure 31 with the current SEAM notation, and in Figure 32 with the proposed SEAM notation. The SEAM modeler used the same graphical conventions (oval for service and rectangle for properties) and the same text labels. The color coding of properties enables readers to identify that the first two service characteristics (replacement car and duration) correspond to Monica's goal (colored with light blue) and that the other characteristics (price) correspond to George's goal (colored with light orange).

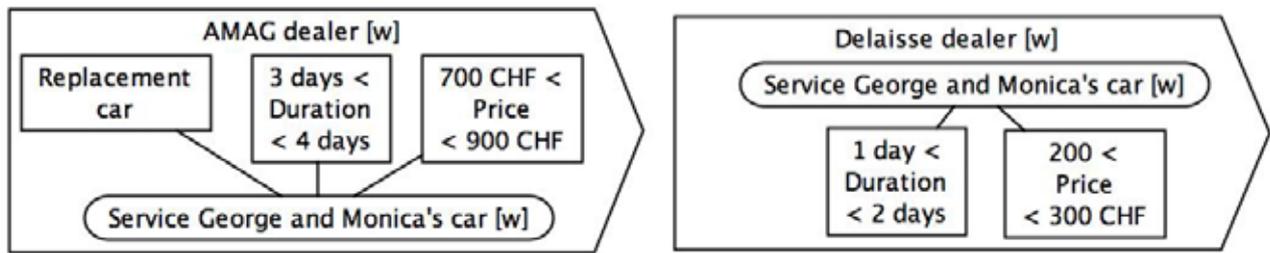


Figure 31: Service offering with the current SEAM



Figure 32: Service offering with the proposed SEAM

In our models, all properties are identified with an icon. For the replacement car, the picture with a parking lot of Audi replacement cars was used. For the duration of 3-4 days (AMAG), a calendar icon with the period marked with two red dots was used. For the duration of 1-2 days (Delaisse), the two dots are marked closer. For the price, an icon showing a set of banknotes was used. We model the fact that there is no replacement car with text and show this service characteristic explicitly in the model (Figure 32), rather than not show it at all (Figure 31). We also add the “interface” to the customers, or the single-point-of-contact by using the icons of the receptionists: “Marc” for AMAG and “John” for Delaisse. We noticed from readers that by using the photo of a woman instead of a man for AMAG would lead to a possible interpretation of stereotypes (Monica would want AMAG because she would interact with a lady understanding her needs, whereas George would want Delaisse because he would interact with a technician proposing a lower price). The photo of the person chosen is therefore critical for readers to understand the correct implications of the model: for instance, John might be both the receptionist and the mechanic for Delaisse – in this case a different photo should be chosen for each role.

In other model iterations (Figure 33) we explored the possibility of positioning the two dealers to the right and the two customers to the left, but the arrow box became confusing for readers. We also colored the background in grey (to mark wholes) and in white (to mark composites), but readers did not notice the difference. Instead of the photo used as background, we explored using it in smaller size at the top, as for other actors. We used other colors for time, money and replacement car in relation to the customers’ goals and beliefs (green for George, red for Monica, yellow as bonus). These proposals guided readers to develop stereotypical judgement. They thought that red means problem, and therefore Monica’s goal has to be satisfied, whereas green means “ok” – for instance both price ranges proposed by the two dealers were fine for George’s budget. The replacement car was interpreted as a bonus / gift or as part of an extended service that has a certain cost (marked with yellow to suggest attention). These interpretations (problem / not problem / attention) did not correspond to what the modeler wanted to communicate: the relation between the customers’ goals

and how they are satisfied by the dealers' service offerings. However, they led to the proposed SEAM models in which we avoid common color stereotypes through the use of more neutral colors.

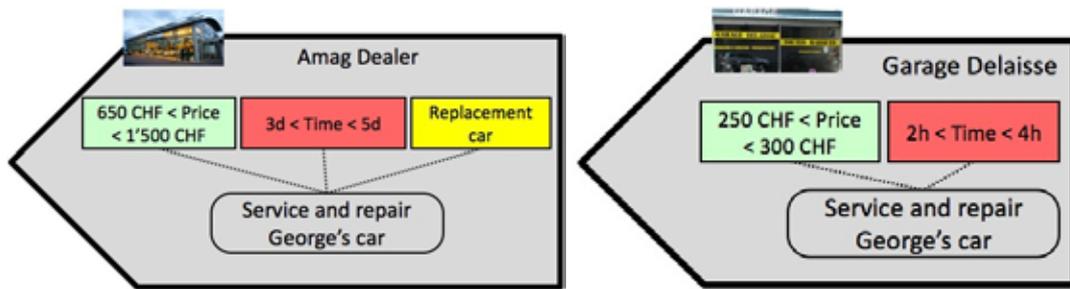


Figure 33: Service offering from model iterations

Terminology

SEAM modelers need to be cautious with the terminology they use. As experts, they could include services for each system as a whole and model the fact that “the system provides a service”. Then, they might show this service as implemented by the corresponding process, together with the component services that participate in this process, thus linking services with processes at the same hierarchical level. In contrast, readers were not able to understand the link between services and processes, hence they were not able to explain the interaction between systems. Therefore, for readers to understand, it becomes important for SEAM modelers to adapt their discourse to the required level of abstraction.

Readers encountered difficulties regarding the usage of the appropriate terminology corresponding to the context that is shown in SEAM, by using the boundary of the system. Instead of using notation-specific vocabulary, such as the word “process”, to mark the lifecycle of a system as a composite, SEAM modelers should use more generic words such as “relationship” when communicating with readers. Similarly, for modeling “wholes” and “composites”, SEAM modelers should not use these domain-specific words when communicating with readers, rather explain that the same entity (e.g., organization) can be seen as both “from outside” (overview) and “from inside” (detail).

Indeed, SEAM modelers should use generic terms, such as “relation”, “connection”, “organization”, “goal” and “belief”. They should not use expert terminology such as “service”, “process”, “whole”, “composite”, “composition”, “the service implements the process” and “the services participate in the process”. The terminology used in the model should correspond to the conceptualization of readers instead of the notation-specific terms used by SEAM experts. Furthermore, SEAM modelers should use the terminology that facilitates the readers' understanding of the story as a whole, rather than get lost in details.

We learned from readers that the terminology needs to be precise. For instance, the goals of the two customers (“car available” and “affordable price”) with the current SEAM (Figure 34) need to be understood in relation to the customers' beliefs. We therefore modeled the beliefs first (“Commuting to work will be longer without a car” and “Some dealers charge more than others”) and then the goals (“I need to commute fast to work” and “I need to service the car”) (Figure 35). The “service” word was eliminated. We modeled the two customers to “own car” together in order to show their common interest in the car, rather than separate services: “drive to work” (shows Monica's interest in using the car) and “check and manage car” (shows George's interest in maintaining the car).

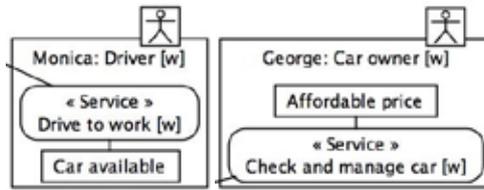


Figure 34: Terminology with the current SEAM

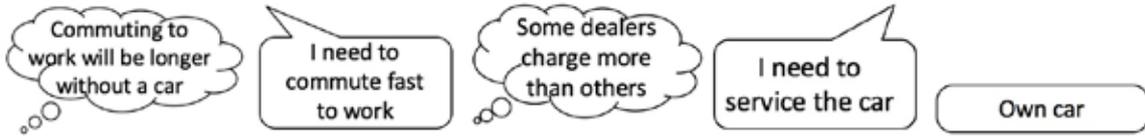


Figure 35: Terminology with the proposed SEAM

In our other model iterations, we explored the degree of concreteness in terminology to express the goals of the two customers. For Monica, we added the text from the first two icons from Figure 36 in which we emphasize the time to commute by car (10 minutes). We mention that this applies to “commute to and from work during weekdays”. For George, we added the text from the last two icons from Figure 36 in which we emphasize the “low price to maintain and repair the car”. We mention that the service needs to be done by a “high quality dealer”.

The 10 minutes’ time, the low price and the high quality dealer were marked in bold. By interviewing readers, we found out that these are too detailed aspects of the story. We simplified the text (no high quality dealer and no commute to and from work during weekdays) and dropped the bold emphasis. For shapes, we used the callout clouds to suggest beliefs and the callout bubbles to suggest goals, because these are graphical conventions readers can easily interpret.



Figure 36: Terminology from model iterations with SEAM

We have shown how the relation with reality was made more explicit in the proposed SEAM model by comparing against the first model iteration and other improvements from other model iterations. Inspired by the readers’ conceptualizations, we have more explicitly shown systems, their behaviors, and their properties. These include actors and their states, their beliefs and goals, fundamental units, and service offering that were represented with appropriate photos, icons and terminology.

4.5.2 Rationale and Choice

We make the rationale and the choice more explicit in the SEAM model. First, we explain the rationale and the choice. Then, we explain how they can be shown, by using the Questions, Options and Criteria (QOC) framework presented in Chapter 2.3.1. Finally, we show how we extended QOC for the SEAM notation.

The rationale considers the reasoning for the design and the evaluation of possible alternative designs. The choice represents the consequence of the rationale or the decision. We document the rationale using visual artifacts that can be easily understood by readers. The Questions, Options and

Criteria framework is one expression used to understand the rationale. Questions refer to “Which one?”, Options provide possible answers to questions, and Criteria help compare options. QOC helps to assess the strengths and weaknesses of different options.

In our example, the rationale refers to the reasons for George and Monica to choose one dealer. The choice refers to which dealer they choose. The choice is based on the assessments of criteria. We analyze the problem, or question, based on the options and the corresponding criteria. In Table 9, at the top, we show the main question (“Which dealer?”). The two options (the two dealers, AMAG and Delaisse) are shown in rows. There are three criteria meaningful for the two actors (price, replacement car and duration).

We add the evaluation of criteria for each actor. The price range proposed by AMAG is evaluated as too high by George, whereas the price proposed by Delaisse is very good. The duration proposed by AMAG is evaluated as too long by Monica, whereas the duration proposed by Delaisse is acceptable. The replacement car is offered only by AMAG and is evaluated as very good by Monica. The fact that Delaisse does not offer a replacement car is evaluated negatively. Other criteria, such as: warranty, proximity, and quality of service are not important factors for the two customers. The assessment of criteria is according to the two actors, George and Monica. For other actors (e.g., of different cultures), other criteria might have been evaluated differently (e.g., the warranty might have a higher importance for certain people). The more complicated the assessment of options is, the more difficult it is for readers to understand the rationale for the actors’ assessments.

Question: Which dealer?

		Criteria					
		Price		Replacement car		Duration	
		George	Monica	George	Monica	George	Monica
Options	AMAG	700-900 CHF 		Yes 		3-4 days 	
	Delaisse	200-300 CHF 		No 		1-2 days 	

Positive | Neutral | Negative

+2 +1 0 -1 -2

Table 9: Question, Options and Criteria for the two customers

The information in Table 9 can be mapped to the Question, Options and Criteria (QOC) framework (Figure 38). QOC allows for only a binary assessment of criteria: either positive (green) or negative (red), for only one actor. We extend this assessment by using a five-level Likert scale (Trochim, 2016): strongly positive (two green happy smileys), positive (one green happy smiley), neutral (one orange neutral smiley), negative (one red sad smiley) and strongly negative (two red sad smileys). The smiley notation is used to quantify the evaluation of each criterion by each of the two customers. It can be mapped to a numerical scale ranging from +2 to -2. Besides this notation we explored using another one based on colored stars or on pluses and minuses (Figure 36). Readers preferred the smiley notation.



Figure 37: Expression of rationale from model iterations

The QOC framework extension from Figure 38 includes the evaluation of each criterion by the two actors, George and Monica, instead of one. We updated the green / red / orange lines between the criteria and the options with the corresponding smileys.

In order to evaluate the two options, the modeler could use an algorithm. One simple algorithm is to compute the sum of all evaluations. In this case, AMAG receives a total of one negative smiley and Delaisse receives one positive smiley. Therefore, the answer to the main question (“Which dealer?”) is: Delaisse. Other algorithms could consider critical criteria: for instance, if the replacement car is a critical criterion for Monica to commute to work, then the option providing it needs to be chosen (in this case AMAG). The choice can be made using “satisficing” (of options, i.e. good enough) and “accommodation” (of conflicting interests, i.e. consensus) when no single option fully meets all criteria of the two customers (Chapter 6.2). Otherwise, if the price or duration would be critical factors, then Delaisse needs to be chosen.

We inscribe the rationale in the SEAM model. Because in SEAM modelers start by modeling the two actors’ goals and beliefs, we swap the QOC representation from right to left in the SEAM model shown in Figure 39. The main question (or problem) is represented with a cloud (with stringent yellow background and red boundary to make it as visible as possible to readers) to the right, the options are represented with red and green hexagons, lines and check marks (for choice) in the middle, and the criteria are represented using smileys (again using stringent yellow background and red boundary to mark the correspondence with the main question and suggest that the answer consists in understanding the relation). We show how the two customers perceive the two dealers’ service offering using the smiley notation: green (positive) and red smileys (negative) on a yellow background (again, to mark the relation with the main question). The red hexagon connects the customers with the non-chosen option, whereas the green hexagon connects the customers with the chosen option. Readers are able to understand the reasoning leading to this choice. This model instance represents the choice, or the decision of the two customers.

We have shown how the choice rationale was made explicit in the proposed SEAM model, based on an extension of the Questions, Options and Criteria framework for two actors (instead of one). The criteria are evaluated based on customers’ perceptions, thus extending QOC (that considers only positive and negative assessments) with a quantitative scale based on (colored) smileys. This quantitative representation was previously used in SEAM Supplier-Adopter-Relationship models with “++”, “+”, “-” and “--” signs. We made this representation more explicit: red (negative), orange (neutral) and green (positive) smileys shown with yellow background.

Design rationale complements Goal-Belief modeling in SEAM, in which goals and beliefs are combined to form choices. It is an alternative to the Supplier-Adopter-Relationship model, which is complicated to show to readers who do not have a SEAM background because it includes the view of the dealer as a composite, which contains service components. In the model instance showing the choice (Figure 39) we model the assessment of features by the customers, together with their values

expressed with goals and beliefs. We add the customers common main goal (the problem), that is not included in SEAM Goal-Belief.

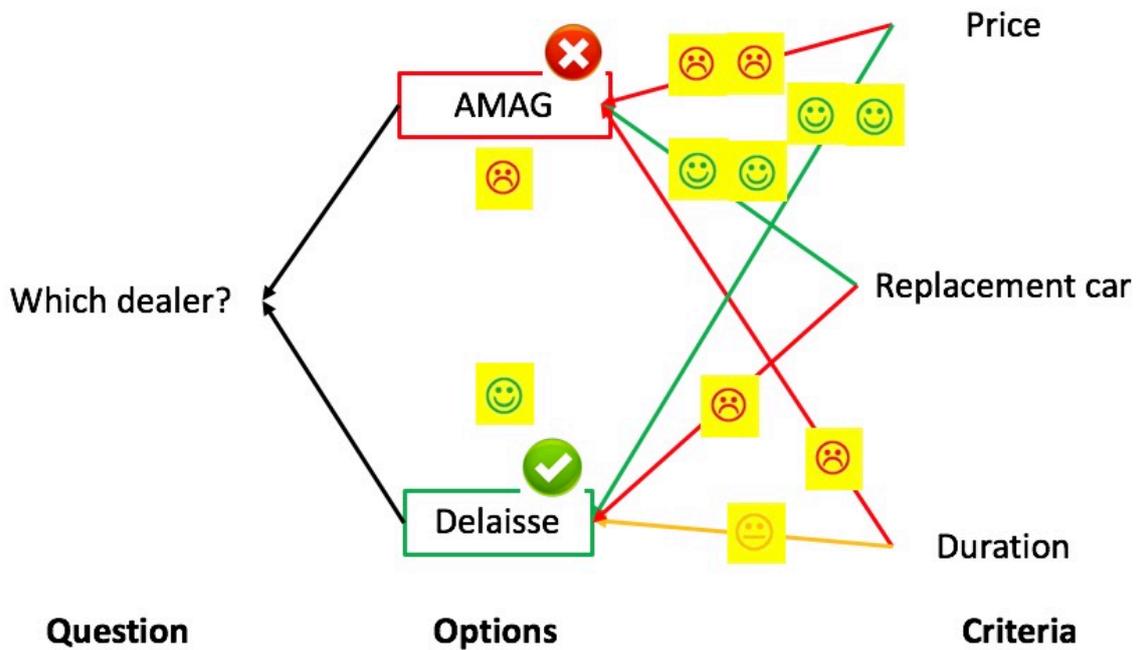


Figure 38: Questions, Options and Criteria framework extension in SEAM

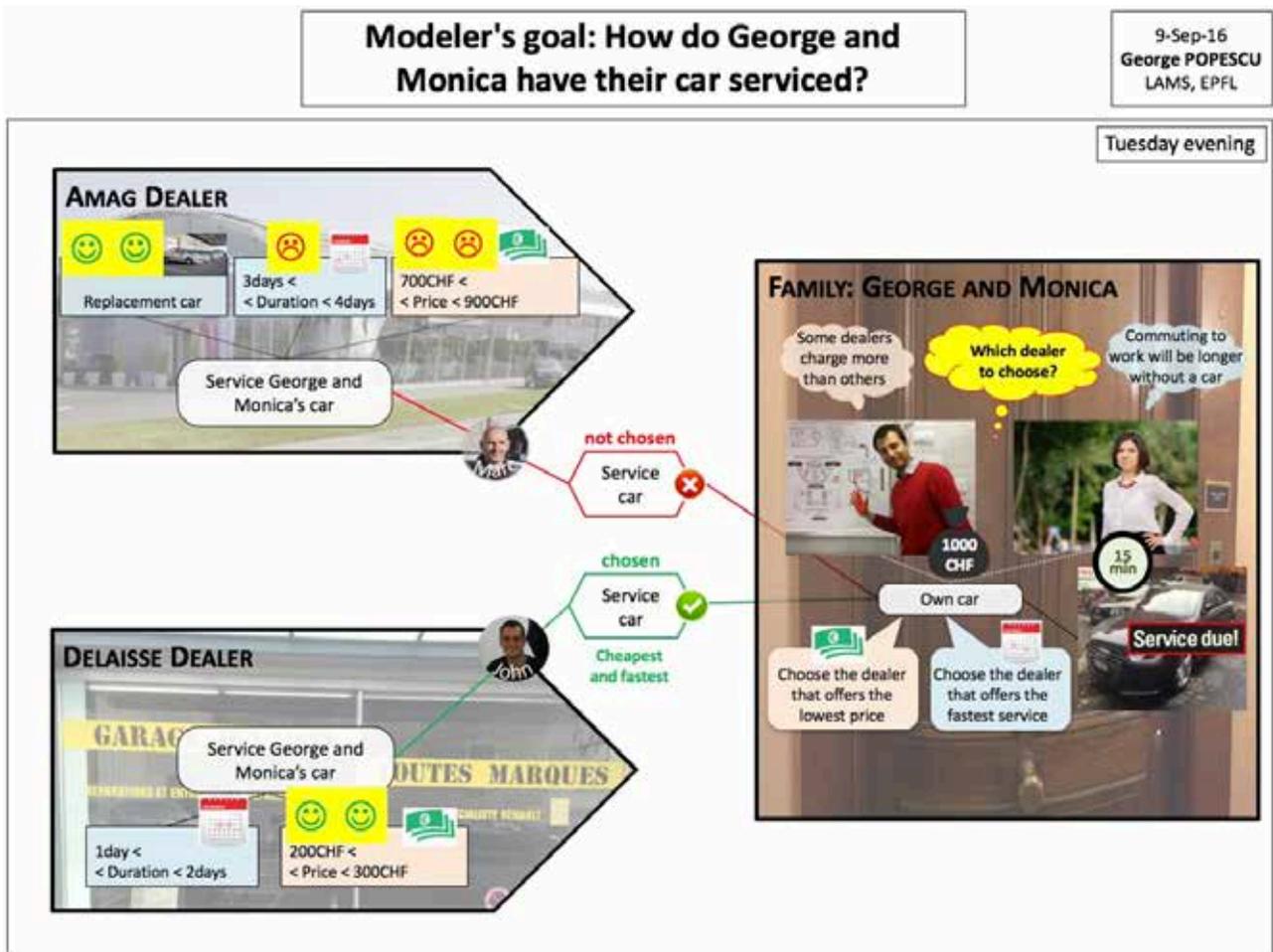


Figure 39: Choice rationale based on Questions, Options and Criteria with the proposed SEAM

4.5.3 Story

We use the four phases (context, conflict, climax and closure) to create a storyboard model. This model includes a sequence of one to three model instances for each story phase, which helps the modeler convey his story (Figure 40).



Figure 40: Storyboard design for each SEAM model instance based on the four story-phases

We create one model instance to model the context and the closure because it suffices to introduce the setting and characters and present the solution to the problem. We create three model instances to model the problem because we want to help the readers understand in detail the rationale leading to the choice of the dealer. We create two model instances to model the climax because we want to show how the car is serviced by the dealer. Each of these story phases are described in detail in this section.

Story-driven modeling aims to provide a systematic process for the creation of story-based models. As numerous possible scenarios can emerge from each model instance, the modeler needs to be cautious about directing the readers' attention towards the key elements in his story, and avoiding unnecessary details.

In organizing the story, the SEAM modeler needs to categorize objects together (e.g., actors, goals, services, features), separate the story in model instances that correspond to each story phase, group elements in model instances, and fill in the gaps between the model instances by using visual cues. Our interviews showed that readers relate well to chronological sequences of events as they want to know "What happens in the model?". The passage of time is shown with a rectangle displaying the date at the top right hand-side of each model instance below. The car-maintenance service duration is two days. Events occur between "Tuesday morning" and "Thursday evening", as indicated on each model instance.

Context

The context is the first part of the storyboard (in orange in Figure 40). It includes the setting and characters. We design the context by using one model instance.

In this first model instance (Figure 41), we show the two actors: George and Monica. They form a family and own a car together. Each has beliefs (shown at the top) and goals (shown at the bottom). The beliefs are related to the actors' realities. George works at EPFL in Lausanne. Monica works at Procter & Gamble in Geneva. The goals are related to each actor's desires, which are distinct. George needs to service the car. Monica needs to commute fast to work. However, they both need to decide upon "how to have their car serviced". This situation occurs on "Tuesday morning", as indicated with the rectangle at the top right.

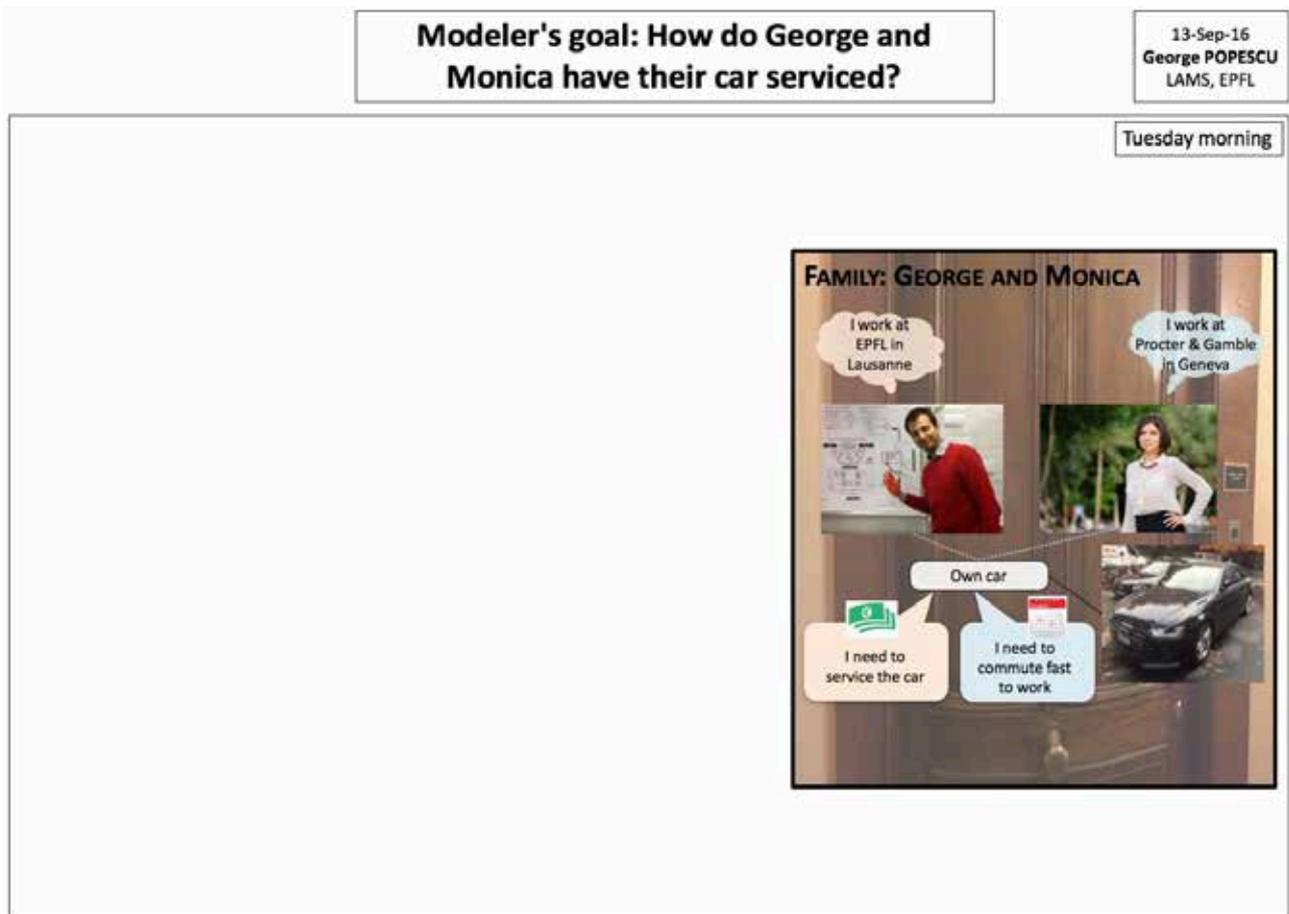


Figure 41: The first model instance of the proposed SEAM model (context)

Conflict

The conflict is the second part of the storyboard (in red in Figure 40). It describes the challenge or problem. We model the conflict using three model instances. We show explicitly the question, options, criteria and their assessments in order to explain the rationale and choice.

In the second model instance (Figure 42), we introduce the problem with the text (“Service duel!”), as displayed by the car’s computer. We also show, on the left, the two Options of dealers (AMAG and Delaisse) with their corresponding service offers (or Criteria). The readers see two numerical values of “1000 CHF” for George (corresponding to his budget) and the “15 minutes” for Monica (corresponding to the commute time by car – any car – to work). Readers notice that George’s belief has changed to “Some dealers charge more than others”. Monica’s belief has also changed to “Commuting to work will be longer without a car”. Now, that the selection has been restrained to only two dealers, George’s goal is: “Choose the dealer that offers the lowest price”, and Monica’s goal is: “Choose the dealer that offers the fastest service”. The two dealers are presented with their service offerings (or service characteristics). AMAG offers a replacement car during the service, proposes a duration between 3 and 4 days, and demands an estimating price between 700 and 900 CHF. Delaisse does not offer a replacement car, but proposes a lower service duration between 1 and 2 days, and demands for a lower price between 200 and 300 CHF. This situation occurs on “Tuesday evening”.

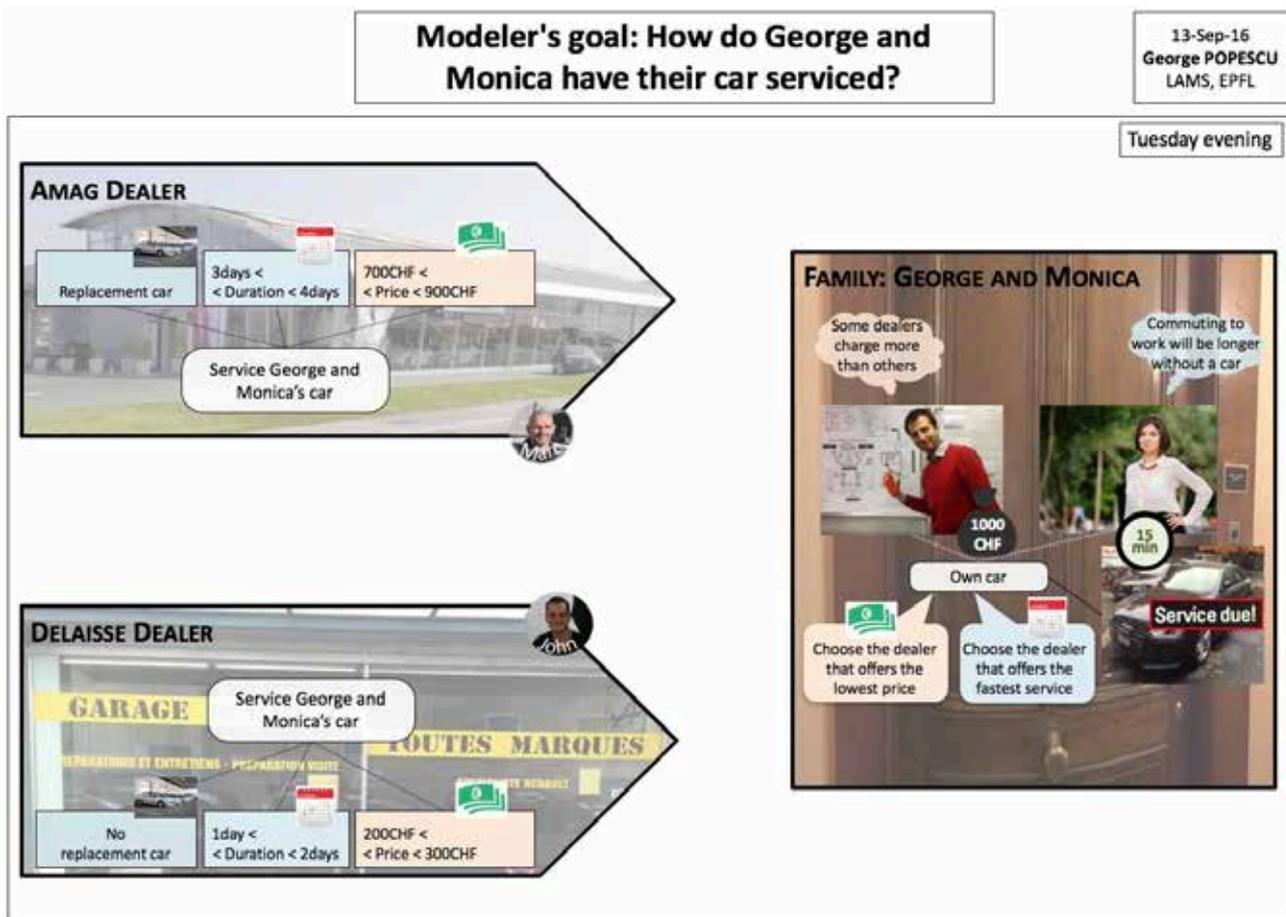


Figure 42: The second model instance of the proposed SEAM model (conflict)

The third model instance (Figure 43) shows the customers' main question ("Which dealer to choose?") together with the assessment of criteria by each of the two customers. The question is placed between George and Monica because it needs to be answered by both, as a family, despite conflicting interests. This assessment is shown with smileys. Their yellow background suggests the association with the main question. The color coding allows readers to perceive that time-related service characteristics are evaluated by Monica (light blue) and price-related ones are evaluated by George (light orange). Therefore, for AMAG, Monica evaluates as strongly positively the replacement car and negatively the duration, whereas George evaluates the proposed service price as strongly negatively. For Delaisse, Monica evaluates the lack of replacement car as negatively and the duration as neutral, whereas George evaluates the proposed price as strongly positively. This situation occurs on "Tuesday evening" as well.

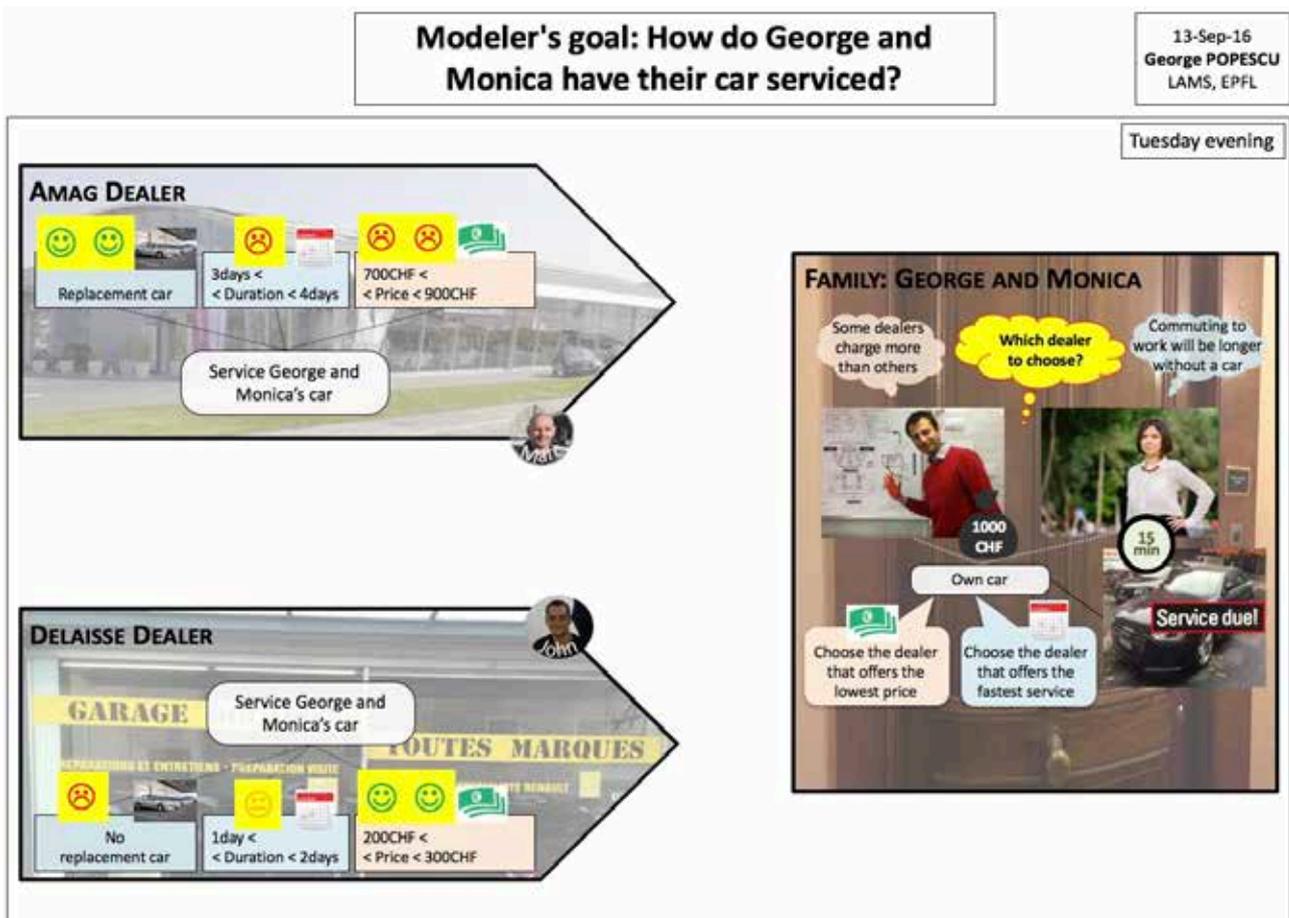


Figure 43: The third model instance of the proposed SEAM model (conflict)

The fourth model instance (Figure 44) shows which of the two options was chosen, as a consequence of the assessment. This is shown with two hexagons colored in red and green, respectively, positioned in the middle of the model instance. The choice for the Delaisse dealer is also made on “Tuesday evening”. None of the two dealers offers the “best option”. In the best case, one of them would have offered a replacement car, a short duration, and a low price. However, one option satisfies (is good enough) the criteria of the two customers: despite the fact that Delaisse offers no replacement car, it does offer a short duration and a low price. Therefore, Monica and George prefer to pay less for a “downgraded service”, without the replacement car (commuting in 35 minutes), proposed by Delaisse, rather than pay more and keep the “same service”, with the replacement car (commuting in 15 minutes), proposed by Delaisse, rather than pay more and keep the “same service”, with the replacement car (commuting in 15 minutes). If commuting would be difficult or critical, then the merits of the replacement car would weigh the choice in favor of AMAG (e.g., if Monica would have an important meeting that he needed to attend in due time). Similarly, if the warranty of the car would be important then the choice would be for AMAG as well (e.g., if servicing the car with Delaisse would invalidate the car’s warranty).

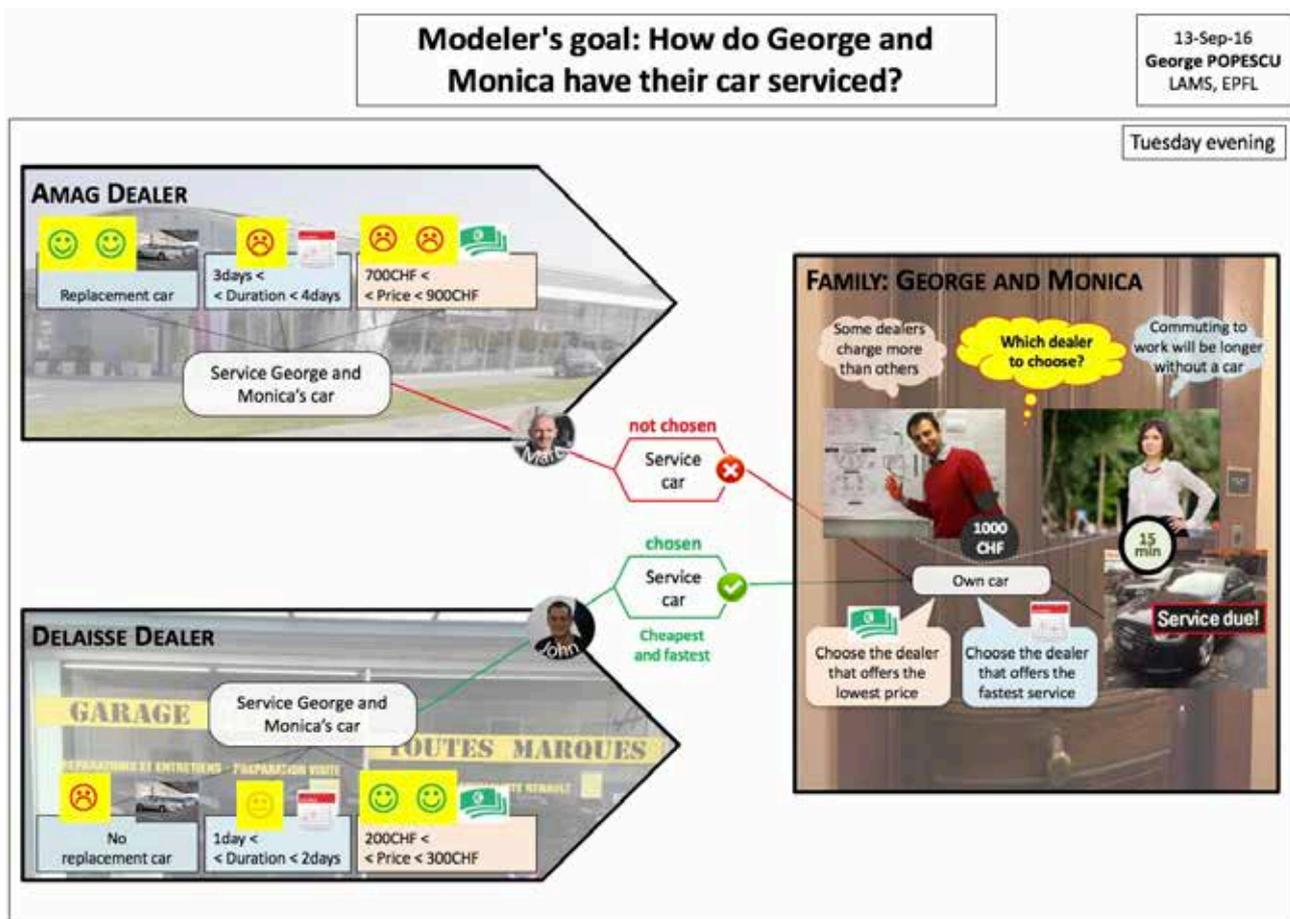


Figure 44: The fourth model instance of the proposed SEAM model (conflict)

Climax

The climax is the third part of the storyboard (in green in Figure 40). It describes the turning point. We design the climax by using two model instances.

The fifth model instance (Figure 45) shows that the two customers have received the dealer's offer and are waiting for the bill. The car is shown at the dealer with the hood open, which means it is being serviced. During the service, the two customers need to commute work. Monica takes the tram, with which the commute time to work is longer (35 minutes) than with the car (15 minutes). George is waiting for the bill as indicated with his goal. The readers notice that his budget has not yet changed. This situation occurs on "Wednesday morning".

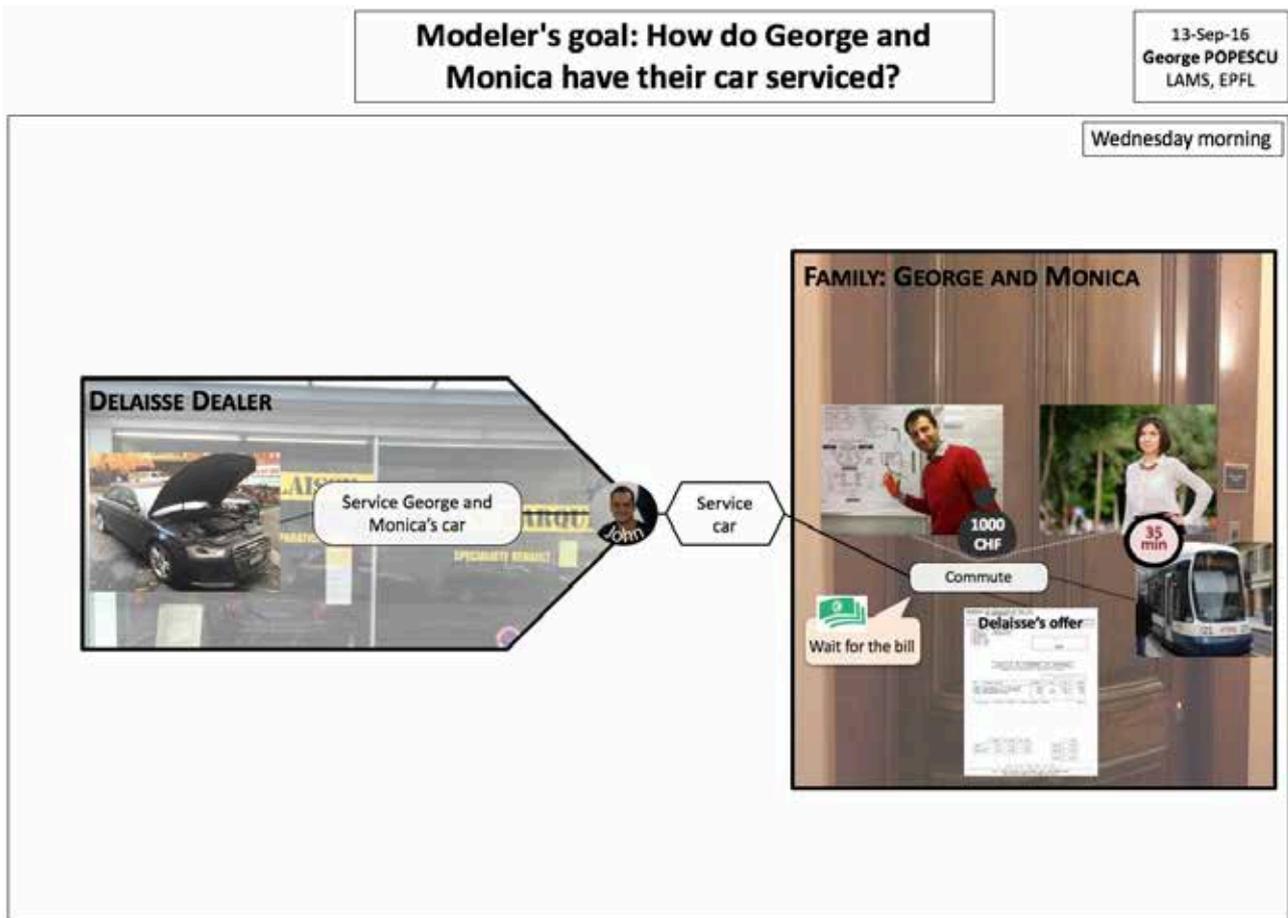


Figure 45: The fifth model instance of the proposed SEAM model (climax)

The sixth model instance (Figure 46) shows the exchange of money for the service. The customers have the bill and their car. They find that “the price was reasonable” – their common belief. The price is deducted from the budget so that George now has “750 CHF” left. The amount paid is shown with 3 banknotes at the dealer (two of 100 CHF and one of 50 CHF). The “service car” hexagon in the middle is no longer colored with green in order to express that the choice was already made (no longer necessary). This situation occurs on “Thursday evening”, two days after the car was left at the dealer.

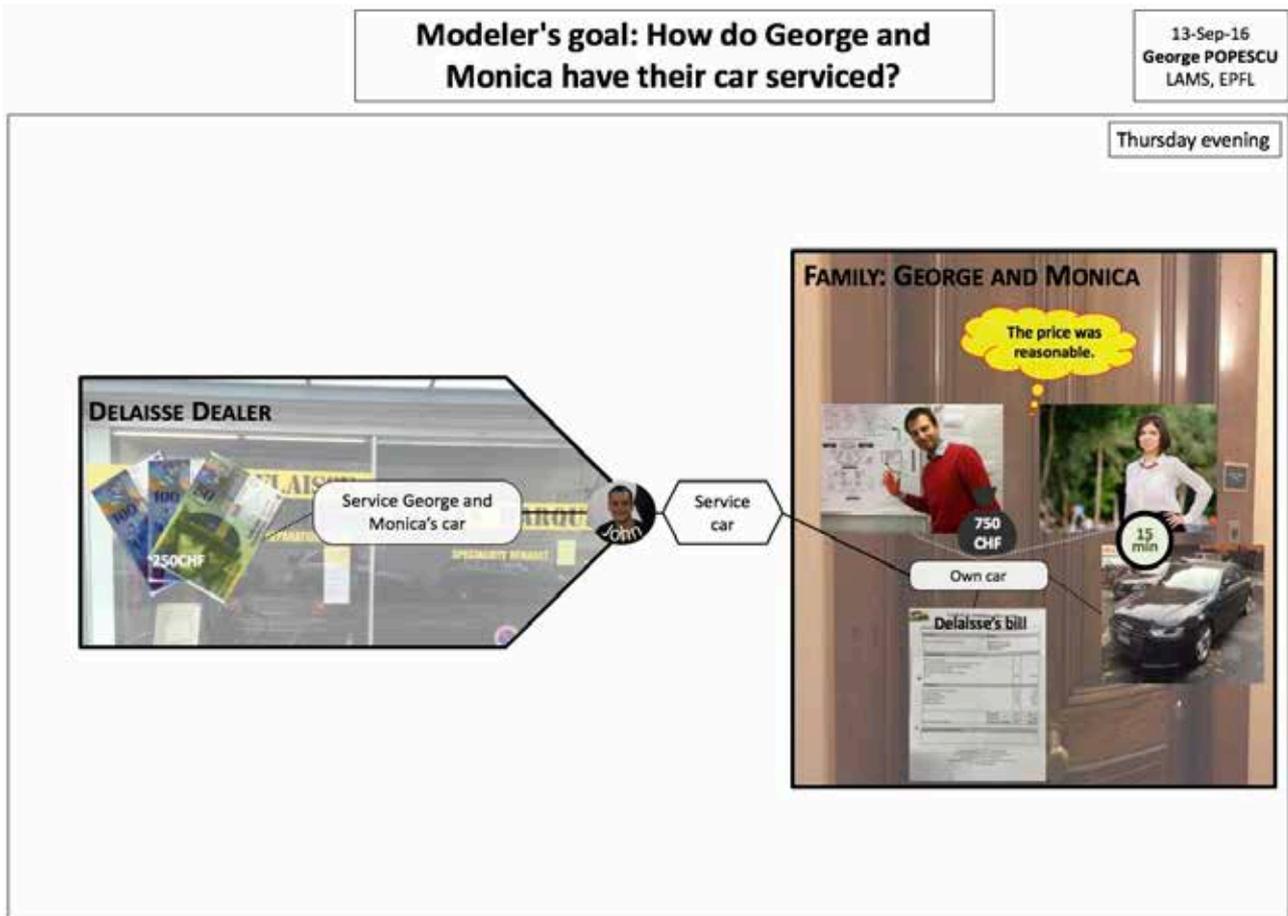


Figure 46: The sixth model instance of the proposed SEAM model (climax)

Closure

The closure is the fourth and last part of the storyboard (in blue in Figure 40). It describes the resolution or the solution to the problem. We model the closure using one model instance.

The seventh model (Figure 47) instance shows the actors with the solution (without the problem). They have the same beliefs as in the first model instance (context story-phase). The car is functional and does not need to be serviced any more (the “Service due!” indicator was removed). This model instance is the same as the first one, shown at the beginning of the story, with the exception of the goals (needs) of the two customers. The goals of the two customers are not shown and, consequently, nor are the car dealers that might solve their problem. This situation occurs on “Thursday evening”.

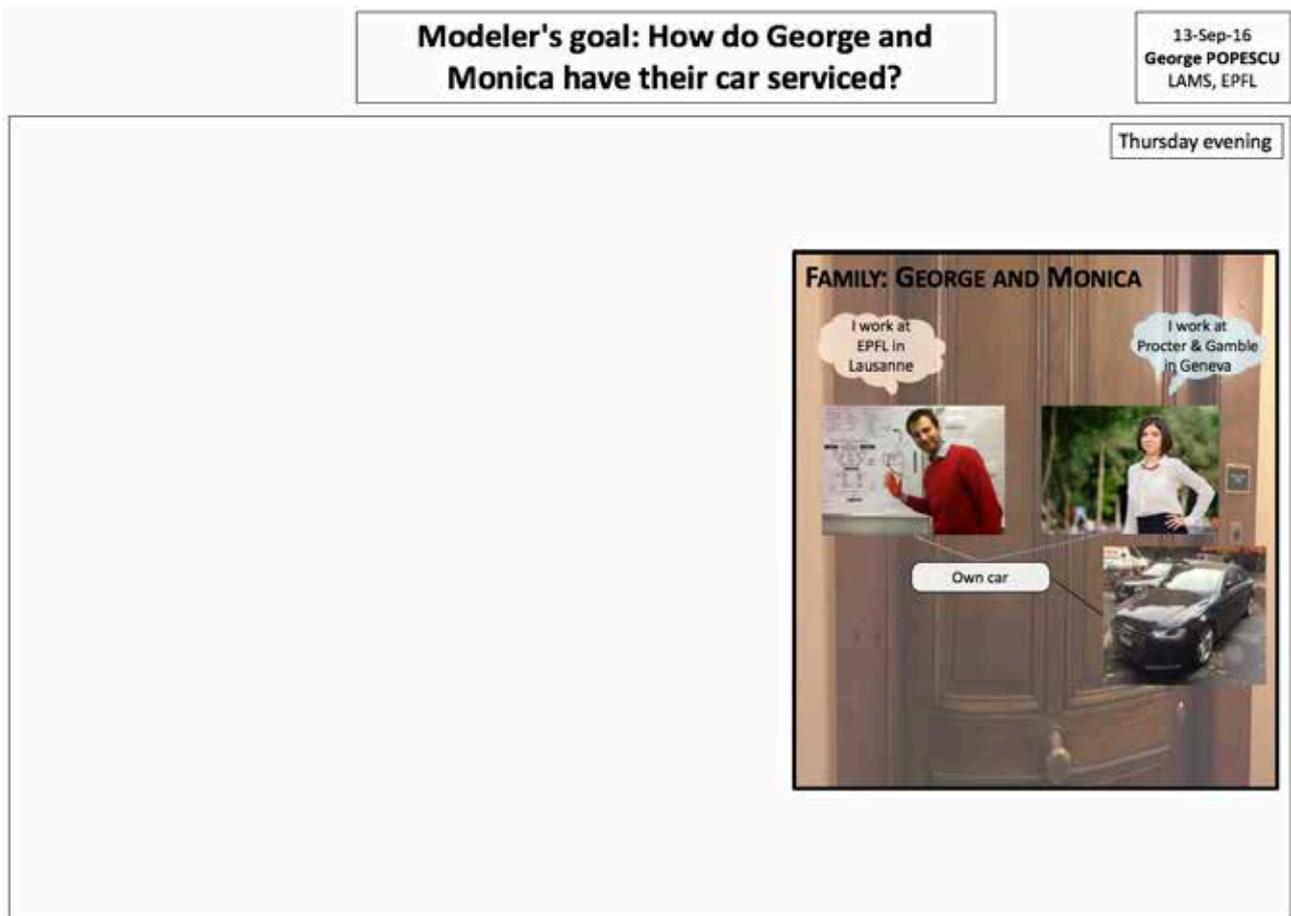


Figure 47: The seventh model instance of the proposed SEAM model (closure)

Concluding remarks

SEAM modelers consider one or two model instances - the current problem(s) and future solution(s) represented with two model instances (“as-is” and “to-be”). We extend SEAM with storytelling, by creating a series of multiple model instances (seven instead of one or two) that show a sequence of (temporal) changes, based on the four story-phases (context, conflict, climax and closure). Each phase consists of one to three model instances that describe the change of actors’ states in the model. Readers can follow this change, as the majority of graphical symbols are kept in place (appear / disappear), with the exception of few (car, money, dealer). Therefore, the redundancy (or repeated emphasis) of graphical elements helps readers understand a certain stability of the model combined with a limited change in the actors’ states. Showing the problem and the solution is not sufficient because the change needs to be explained in more detail in order to be understood by readers. Including a granular division of the sequence would require showing every change of all actors’ states. We focus on the most meaningful changes relative to the story. The modeler needs to analyze the trade-off between the changes and the model’s complexity.

4.6 Summary

In Chapter 4, we have presented our research design (Chapter 3.4) applied to evaluate models created with the SEAM notation. We have shown how to make notation elements more explicit so that readers understand the story. We have created a proposed SEAM car-service maintenance model with seven model instances (presented in Chapter 4.5.3). Our contributions to SEAM modelers are three-fold:

- We model more explicitly the *relation with reality* to show the modeler’s goal and model information, actors (human, objects and organizations), and the service offering, by using photos, icons and terminology that correspond to the readers’ conceptualizations
 - Include perceived real-life occurrences of objects (e.g., pictures of George and Monica, AMAG and Delaisse, car, service due indicator, tram, money, offer, bill) to show concreteness
 - Include instances of objects (e.g., money and time metaphors) that readers can relate to
- We model the *rationale* (why the problem is meaningful) and the choice (the decision) of actors by using the main question, the two options (of dealers), the three criteria (service offering) and their evaluation (using the sad-neutral-happy smiley notation)
 - Model customers’ rationale by showing their goals and beliefs that contribute to their evaluation of criteria
 - Model customers’ choice as a consequence of their rationale
- We model the *story* by using the context, conflict, climax and closure story-phases to show how the problem (George and Monica need to service their car) is solved (the car is serviced)
 - Model actors’ states to show change (e.g., car with the hood closed / open, car with the customers / at the dealer)
 - Include indications about the period of the day on each model instance (e.g., from Tuesday morning to Thursday evening)

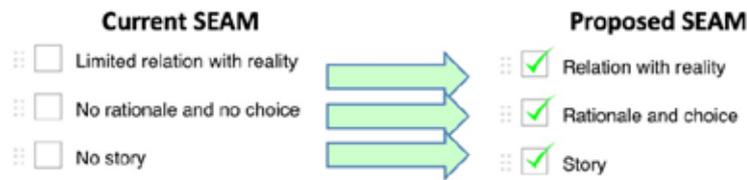


Figure 48: Contributions to SEAM - from the current SEAM to the proposed SEAM

Instead of creating three distinct views of a situation using SEAM Goal-Belief, SEAM Behavior and SEAM Supplier-Adopter-Relationship models, we propose one model, constructed using story-phases, that shows a coherent view. The model presented in this chapter represents our best proposal after interviewing one hundred readers. It is not the best in absolute terms. The main part of the research work consisted in identifying what is important to be shown in the model and how to make the story clear for readers.

The proposed model includes some actors' (the customers) goals and beliefs (instead of all actors) from the SEAM Goal-Belief model, the services offered by the dealers connected to the process life-cycle, shown as choice (instead of all services and processes) from the SEAM Behavior model, and the features (at the suppliers), values (perceived by customers) and their evaluation (expressed with smileys) from the SEAM Supplier-Adopter-Relationship model.

The SEAM notation is permissive (or adaptive) in using photos, icons and terminology and a wide variety of visual variables (color, value, texture, shape, size, orientation). Modelers need to pay attention to the principles underlying the notation (e.g., modeling systems seen as wholes into a system seen as a composite, show the process for the system as composite and services for each subsystems as wholes, link services to processes, model goals in relation to beliefs, model features in relation to values).

These contributions were based on readers' feedback and were validated by SEAM experts. The following reactions are from one of the readers who evaluated the proposed model: "This is what they want...", "This is the budget.", "This is the time to commute by car.", "These are the two options...", "This is the problem: what to choose.", "Based on the criteria of lowest price and fastest service they choose the Delaisse dealer.", "Oh yeah, so here is actually the decision.", "Ok... so now they decide for the second option...", "This is the time to commute by tram.", "On this date this is what happens...", "This is the final price paid.", "Now they have the car back after only 2 days and 250 CHF.", "This makes sense!". The reader's interpretation of the model corresponds to the modeler's story. Audio recordings of participants' reactions were transcribed in order to structure feedback.

Chapter 5 Application to i^*

We show how we applied the research method to improve the i^* notation, using the meeting scheduler example presented by (Yu E. S., 1997). i^* already includes a partial rationale. We model more explicitly the relation with reality, the rationale and the story. In brief, the story is the following: “What are the characteristics of a scheduler desired by the meeting initiator and participants?” The i^* modeler asks himself: “How do I, as an i^* modeler, create an i^* model so that readers understand the story that I want to convey?”. We propose improvements for the i^* model.

5.1 Research Protocol

We instantiate the steps of the research design from Chapter 3.4 as follows:

1. Use the initial model presented by (Yu E. S., 1997) in which a meeting initiator and participants agree on the characteristics of a meeting scheduler
2. Interview readers to evaluate how they understand the story that the modeler wants to tell
3. Collect suggestions on how to improve the model and the notation
4. Implement suggestions in a new model iteration (while maintaining the identity of the i^* notation)

It was necessary to interview twenty readers and create twelve model iterations by repeating steps 2-3-4 until the modeler’s story was correctly interpreted by readers. Therefore, on average, between one and two readers evaluated one model iteration. With the last model iteration, readers could interpret the modeler’s story correctly. These readers did not evaluate any previous iteration. Some iterations of the i^* model, the ones that show most meaningful changes compared to the previous iterations, are included in Appendix 2.

5.2 Research Design

The five components of the research design (Chapter 3.2) applied to i^* are:

- Study question: “How should an i^* modeler create a model so that readers understand the story that he wants to tell?”
- Propositions: i^* model iterations from the original model presented by (Yu E. S., 1997) to the last model iteration presented in Chapter 5.5.3.
- Unit of analysis: Readers who evaluated and gave feedback on i^* model iterations.
- Logic linking the data to the propositions: Readers’ interpretations of the modeler’s story.
- Criteria for interpreting findings: Improvements from one model iteration to the next, as perceived by readers, with respect to the modeler’s story. According to the notation designer, model iterations need to maintain the identity of the notation.

5.3 Interviews

We use the example of the following story: “What are the characteristics of a scheduler desired by the meeting initiator and participants?”. We are interested in creating an i* model, so that readers understand this story. The meeting scheduler case is presented in (van Lamswerde, Darimont, & Massonet, 1995).

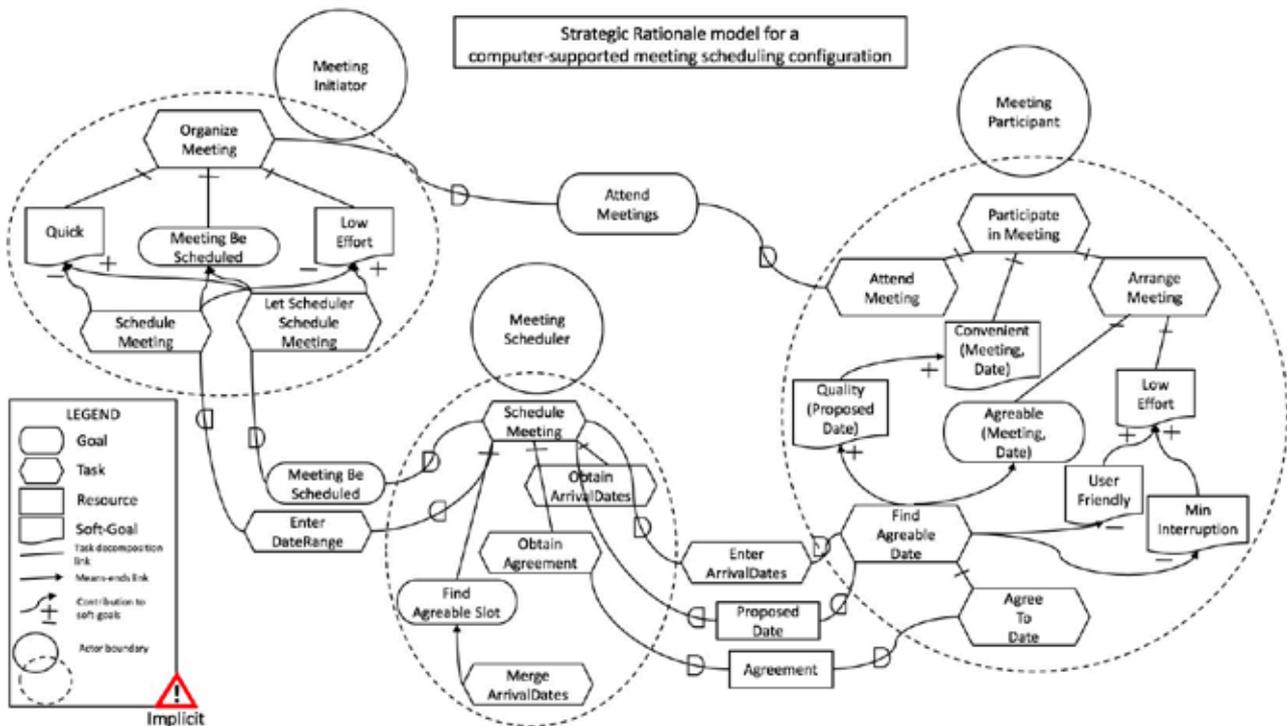


Figure 49: i* initial model – source: (Yu E. S., 1997)

The initial model is presented in (Yu E. S., 1997) and shown in Figure 49. In i*, the modeler focuses on high-level business needs by analyzing strategic relations between actors, based on their goals, beliefs, tasks and resources.

According to (Yu E. S., 1997), this i* Strategic Rationale model shows “the meeting scheduling with a computer-based meeting scheduler”. The full description was presented in Chapter 2.1.2.

In short, a meeting initiator wants to schedule a meeting with participants. For this, he relies on a meeting scheduler to facilitate meeting scheduling. The initiator proposes meeting dates and locations that participants accept via the scheduler. We are interested in showing the characteristics of the scheduler based on the meeting initiator and the participants’ desires about the features of the scheduler. For the initiator, it is important that the scheduler is easy to use, quick, and capable to send a meeting confirmation. For the participants, it is important that they can choose different slots and access meeting details.

As for SEAM, readers need the text and the legend to understand the model. They also need the definitions of actors, goals, softgoals, tasks, resources and relations. But this is not enough. It is difficult for modelers to correctly and fully write the textual description of the model. It is also challenging to create an audio or video narration to explain all the intricacies of the model.

5.4 Feedback

Understanding the i^* model requires knowledge about the i^* notation. We highlight the elements that are difficult to comprehend (implicit) by readers. We group them into three sections: relation with reality, rationale and story.

5.4.1 Limited Relation with Reality

Interviewees were confused by the concepts used in i^* , such as “softgoals”, “resources” or “means-end link”. They were not sure if their interpretation of the model was correct. They did not know what the pluses and minuses on some links mean, and why the information flows from one shape to the next.

The title of the model was confusing to readers as they don’t know what the terms, such as “strategic rationale” or “scheduling configuration”, mean. Readers searched for visual cues to identify where to start interpreting the model from.

Another example is that readers found difficult to interpret the fact that the main task can be decomposed into a goal, a softgoal or a task. They observed that there can be multiple participants but only one is shown in the model. Similarly, there can be multiple date proposals suggested by the initiator and multiple preferred dates suggested by the participants.

5.4.2 Partial Rationale

The initial model does not explain the rationale leading to the characteristics of the actors’ rationale of the characteristics of the meeting scheduler. As observed by readers, if the meeting participants do not attend the meeting, the initiator will not achieve some goal. This goal is not made explicit in the model. In order to schedule meetings, the initiator depends on the participants to provide information about their availability (preferred dates). In order to arrive at an agreeable date, the participants depend on the initiator to accept proposals. Once a meeting is proposed, the initiator depends on the participants to indicate whether they agree with the date or not.

5.4.3 No Story

Similar to SEAM, the single instance of the model does not enable readers to understand how to interpret the story because it represents only one moment in time. Readers were confused as they could not identify a sequence of instances.

It was unclear for readers the order in which they could interpret the model. They did not identify a starting point, but only looked at the IT-specific terminology in the model. The story did not emerge from the model.

The readers’ initial reactions to the model from Figure 49 show little comprehension of the modeler’s story. One participant interpreted the model as follows: “I think you want to model a process of scheduling meetings between a meeting initiator, what he needs to do. Basically he wants to do something quick and with a low effort. He uses a scheduler. And then I think you want to represent what a meeting scheduler does: schedules the meeting, obtains arrival dates, obtain agreement, find agreeable slots, and merge arrival date and... what the meeting participant ... he wants to participate

in the meeting, so attend the meeting, and to arrange the meeting. It needs to be convenient. But it is a bit difficult, I think, to read it. There are some pluses and minuses here, which I need a bit of your help to understand what they mean.”

As this example shows, readers read the words shown in the model but do not internalize the meaning of the model. Their reactions demonstrate little comprehension: “I’m a bit confused still about the meeting scheduler. So it is like a system... Does it have a task or ...? This is the piece that I am most confused about...” As the model improved, we noticed specific reactions that show a better comprehension. For instance, readers asked questions about the details of the graphical symbols rather than their meaning (e.g., Why the question mark shown on the scheduler changes to a tick-mark; what do the red and green linkages suggest).

5.5 Contributions to i* modelers

We present the contributions to i* modelers. These contributions fall under three main areas:

- *Relation with reality* by showing more explicitly the modeler’s goal and model information, actors (human and IT), their goals, tasks, sub-tasks, softgoals and resources (by using the readers’ conceptualization);
- *Rationale* by adding the main question and showing more explicitly the option, the criteria and the assessment of the criteria;
- *Story* by showing model instances corresponding to each story phase: context, conflict, climax and closure.

For each contribution, we highlight the improvements taken from the initial and final model iterations, as well as from model iterations presented in Appendix 2. The evaluation of each model iteration by the participants allowed the modeler to improve the way he showed the relation with reality, the rationale and the story.

5.5.1 Relation with Reality

We show how to make more explicit the relation between the model and the reality that the modeler observes by (i) showing the modeler’s goal and model information, (ii) modeling actors (humans and IT), and (iii) modeling their goals, tasks, sub-tasks, softgoals and resources, thus corresponding to readers’ conceptualization of the situation instead of the modeler’s conceptualization. We compare the initial against and the proposed model iterations and clarify elements from other iterations.

Modeler’s goal and model information

The modeler’s goal is made explicit by including it at the top of each model instance. i* models do not show the modeler’s goal. Hence, i* modelers need to explain it orally when presenting models to audiences. By including the modeler’s goal on each model instance (Figure 50) readers can see the main idea or summary of the story. Each model instance also includes other model information: the date when it was created, the name of the modeler, and his affiliation.

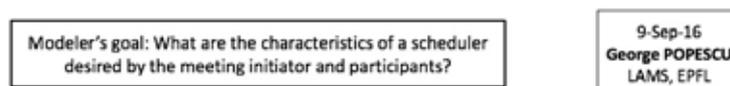


Figure 50: Modeler’s goal and model information with the proposed i*

Actors

There are two categories of actors used in the proposed i^* model: humans (the meeting initiator and meeting participants) and IT (the meeting scheduler). These are shown in Figure 51 with the current i^* notation and in Figure 52 with the proposed i^* notation.

First, we chose to model only one meeting initiator and multiple meeting participants. We chose representative pictures of the actors. For the initiator, we choose the photo of a person sitting in front of a computer. For the participants, we choose the photo of a group of people sitting at a table and discussing. We show the scheduler with a transparent calendar icon and a question mark / check mark. This suggests the search for the scheduler's characteristics and their acceptance (or validation) by the human actors.

We choose a different color boundary for each actor: blue for the initiator, red for the participants, and green for the scheduler. We needed to only distinguish between the three and therefore there was no need to consider particular colors and their meaning.



Figure 51: Human and IT actors with the current i^*



Figure 52: Human and IT actors with the proposed i^*

In our model iterations, we explored pictures of generic and specific actors (Figure 53). The pictures need to be appropriate for the context that is shown. For instance, since the modeler did not consider a specific meeting initiator, we used the icon of a stickman. Similarly, the modeler did not consider a specific group of meeting participants; hence we used pictures of a group meeting (classroom with participants connected to computers, and of a group of people sitting at a table and applauding the success of the meeting, respectively).



Figure 53: Human and IT actors from model iterations

We explored the use of a calendar / agenda photo with no transparency to show a generic example of a meeting scheduler. This examples becomes specific when the initiator and the participants finish the specifications of the requirements of the scheduler, in our case show an example of a meeting scheduler: Doodle. The red border means that the scheduler is not yet specified whereas the

dark blue border means that it is. These explorations from other model iterations allowed us to conclude that readers prefer the photos and icons from Figure 52.

Goals, tasks, sub-tasks, softgoals and resources

Goals, tasks, sub-tasks, softgoals and resources refer to the characteristics of the actors. They are shown in Figure 54 with the current i^* notation and in Figure 55 with the proposed i^* notation. We proposed other graphical symbols for goals (call-out bubble instead of oval), tasks and sub-tasks (thicker to a thinner boundaries), softgoals (cloud bubble instead of script shape) and resources (rectangular parallelepiped instead of rectangle).



Figure 54: Goals, tasks, sub-tasks, softgoals and resources with the current i^*



Figure 55: Goals, tasks, sub-tasks, softgoals and resources with the proposed i^*

We propose a metaphor to be added to each shape in order to suggest goals, tasks, sub-tasks, softgoals and resources. Sub-tasks can be distinguished from tasks via the size of the icon. Goals are similar to softgoals as they are represented with an arrow and a target. However, they are different because goals use only one color of the target showing one major achievement desired by the actor while softgoals have multiple colors on the target and multiple layers showing that these softgoals represent different layers of analysis with respect to goals. Finally, resources are represented with a black spinning wheel and a circular arrow to show their contribution.

The color coding of actors in relation to their goals, tasks, sub-tasks, softgoals and resources enables readers to identify what characteristic corresponds to which actor. The meeting initiator and its characteristics are colored in light blue, the meeting participants and their characteristics are shown in light red, and the meeting scheduler and its characteristics are colored in light green. The scheduler does not have goals or softgoals as it is an IT actor that cannot produce judgements.



Figure 56: Goals, tasks, softgoals and resources from model iterations

Besides the proposed graphical symbols (Figure 55) we also tested other shapes for goals, tasks, softgoals and resources, their colors and border thickness with readers (Figure 56). The first set of graphical symbols (shown to the left) was proposed by (Moody, Heymans, & Matulevicius, 2009). The second one (shown to the right) is inspired by the SEAM notation (goals, actions, beliefs and properties). None were appreciated by readers, who found the proposed notation to correspond much better to their interpretation. The use of a generic shape in conjunction with a pictogram at the top guides readers to understand each element.

Terminology

i* modelers need to be cautious with the terminology they use. As experts, they could include IT-specific terms and show functions with parameters, e.g., “meeting be scheduled”, “quality (proposed date)” and “min interruption”. In contrast, readers did not understand the difference between “organize meeting” and “schedule meeting”. They did not understand the differences between goals and softgoals, and the connections, such as “means-end link” and “contribution to softgoals”. Therefore, for readers to understand an i* model, it becomes important for i* modelers to adapt their terminology to the required level of abstraction.

This terminology used by i* modelers should correspond to the conceptualization of readers instead of the notation-specific terms used by i* experts. Instead of using an i* notation specific terminology, i* modelers should use more generic words and icons that suggest meaning and are closer to common sense. Explicit i* concepts can be “hidden” under generic terms that are understandable by readers. Indeed, i* modelers should use such terms as: “propose / accept”, “organize / participate”, “locations”, “dates”, “confirmation”, “simple interface” and remove expert terminology such as the terms previously discussed.

We learned from readers that the terminology needs to be precise. For instance, the goals of the two human actors (“organize meetings” and “participate in meetings”) with the current i* (Figure 49) need to be understood in relation to the problem they try to solve, their tasks, and soft-goals. We therefore modeled the problem first (“What scheduler?”) for both actors, then their tasks (same terminology as for goals, but using different symbols), and soft-goals (“easy to use”, “receive confirmation”, “quick”, “choose different slots”, and “access meeting details”).

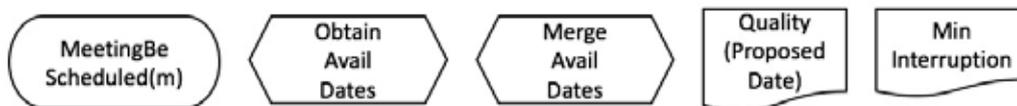


Figure 57: Terminology with the current i*

We used a simplified terminology (Figure 58) in our proposed model compared to the terminology used in the original i* model (Figure 57). This terminology is more straight-forward, does not contain ambiguous terms, parameters of functions or abbreviations.



Figure 58: Terminology with the proposed i*

In our other model iterations, we explored the degree of concreteness in terminology to express the graphical elements related to the three actors (Figure 59). For both human actors we used the question “What meeting scheduler?”, for the initiator we used the second and third icon and terminology to express his needs. Similarly, for the participants we used the fourth and fifth icons and terminology to express their needs. For the characteristics of the scheduler, we described them as, for instance, “simple, easy to use interface” and “quick meeting setup process”. We aimed to reduce the expressions used in the model as much as possible and make the model as concrete as possible.



Figure 59: Terminology from model iterations with i^*

We have shown how the relation with reality was made more explicit in the proposed i^* notation. Inspired by the readers’ conceptualizations, we have more explicitly shown actors and their characteristics. These include human and IT actors, their goals, tasks, sub-tasks, softgoals and resources represented with photos, icons and terminology.

5.5.2 Rationale

We make the rationale more explicit in the i^* model. i^* includes a partial rationale in the model’s construction, but it is not shown sufficiently clear for readers to understand. Below, we show the rationale in more detail, similar to the approach used for SEAM.

Question: What scheduler?

		Criteria									
		Ease of use		Meeting setup process		Confirmation		Different slots		Meeting details	
		Initiator	Participants	Initiator	Participants	Initiator	Participants	Initiator	Participants	Initiator	Participants
Option	Generic meeting scheduler	Easy to use 		Quick 		Receive confirmation 		Choose different slots 		Access meeting details 	

Positive | Neutral | Negative

+2 +1 0 -1 -2

Table 10: Question, Option and Criteria for the meeting initiator and participants

In the considered example, the rationale refers to what the meeting initiator and the participants desire from a meeting scheduler. Table 10 shows the criteria for a generic meeting scheduler that stakeholders consider. Both the meeting initiator and the participants want to attend meetings. For this, they formulate criteria that the meeting scheduler needs to satisfy. The meeting initiator defines the three criteria (ease of use, meeting setup process, and confirmation), whereas the participants define other two (choose different slots and access meeting details). Each evaluates the criteria, based on the five-level Likert scale used in the proposed QOC, in order to define what the characteristics of the meeting scheduler should be.

As for the case with SEAM, the information in Table 10 can be mapped to the Question, Options and Criteria (QOC) framework (Figure 38). We use the five-level scale from strongly positive to strongly negative shown with the smiley notation. The QOC framework extension from Figure 61 includes the evaluation of three criteria by the meeting initiator and of two criteria by meeting participants. There is no need to use an algorithm to compute the score for the scheduler as there is no choice to be made. Rather, the modeler shows that only three of the five desires of the initiator and participants are satisfied by the scheduler. This represents a “satisficing” option that can “accommodate” the evaluation of the initiator and participants’ desires (Chapter 6.2).

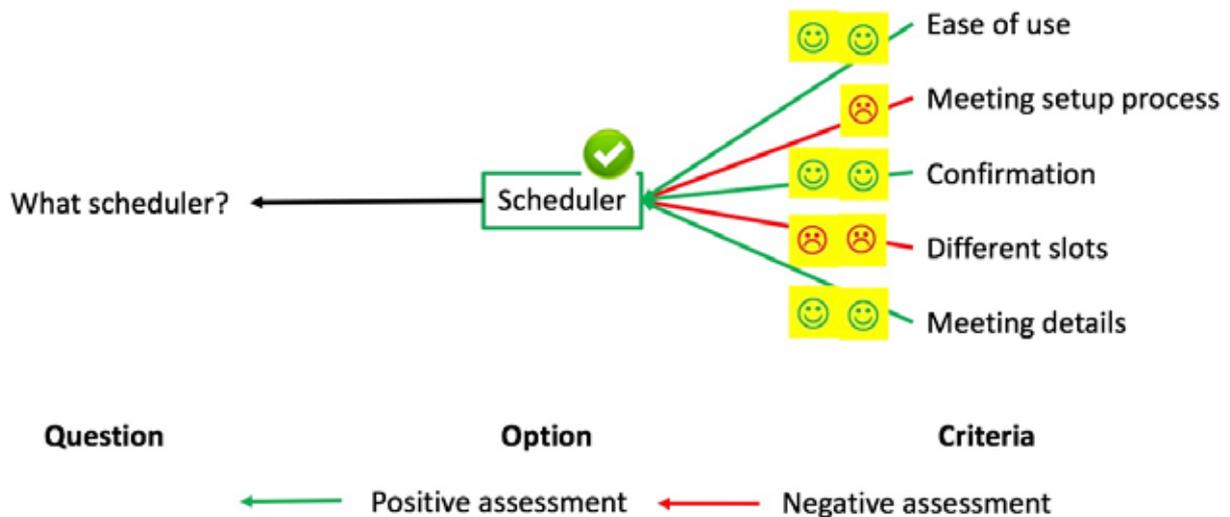


Figure 60: Questions, Options and Criteria framework extension in SEAM

The meeting initiator and the participants specify the characteristics of the meeting scheduler that they would like it to have. In the proposed i^* model (Figure 61), we added the Question (or problem), represented with a yellow oval and red boundary (“Which scheduler?”) for both the meeting initiator and the participants. The Option is the meeting scheduler represented in the middle with green. The Criteria are represented with blue and red clouds corresponding to actors’ softgoals. The assessment of the criteria is binary (like with the original QOC framework) and is shown with the two green and red icons on each line, because the connections between softgoals and resources show the elicitation of requirements from the initiator and the participants. The smiley notation can eventually be used to mark the degree to which each criteria is satisfied by one specific scheduler (e.g., Doodle).

The newly added graphical symbol (oval with red text and yellow background) is not included in the i^* notation and represents an extension. In order to answer the main question, both the meeting initiator and the participants need to define their goals, tasks, sub-tasks and softgoals, by following the i^* decompositions. Then, the softgoals can be linked to the characteristics of the meeting scheduler. We made more explicit the correspondence between the actors’ softgoals and the characteristics of the scheduler using direct links between these softgoals and the scheduler’s resources. Connections are shown differently between goals and tasks, tasks and sub-tasks, tasks and softgoals, and softgoals and resources. The modeler used a simple black line to connect the goal with the main task and the main task with the sub-tasks, for each actor. The main task is connected with bubble links to softgoals. Softgoals are linked with thicker lines to resources. The positive linkages (green happy smileys) between the human actors’ softgoals and the scheduler’s resources are marked with green lines and green icons. The negative linkages (red sad smileys) between the human actors’ softgoals and the scheduler’s resources are marked with red lines and red icons.

We have shown how the rationale was made more explicit in the proposed i^* model, based on an extension of the Questions, Options and Criteria framework for two actors. We have added the question and improved the relations between goals, tasks, sub-tasks, softgoals and resources. The criteria are weighted based on actors’ softgoals with a binary scale (red and green lines) that can be weighted with one / two smileys.

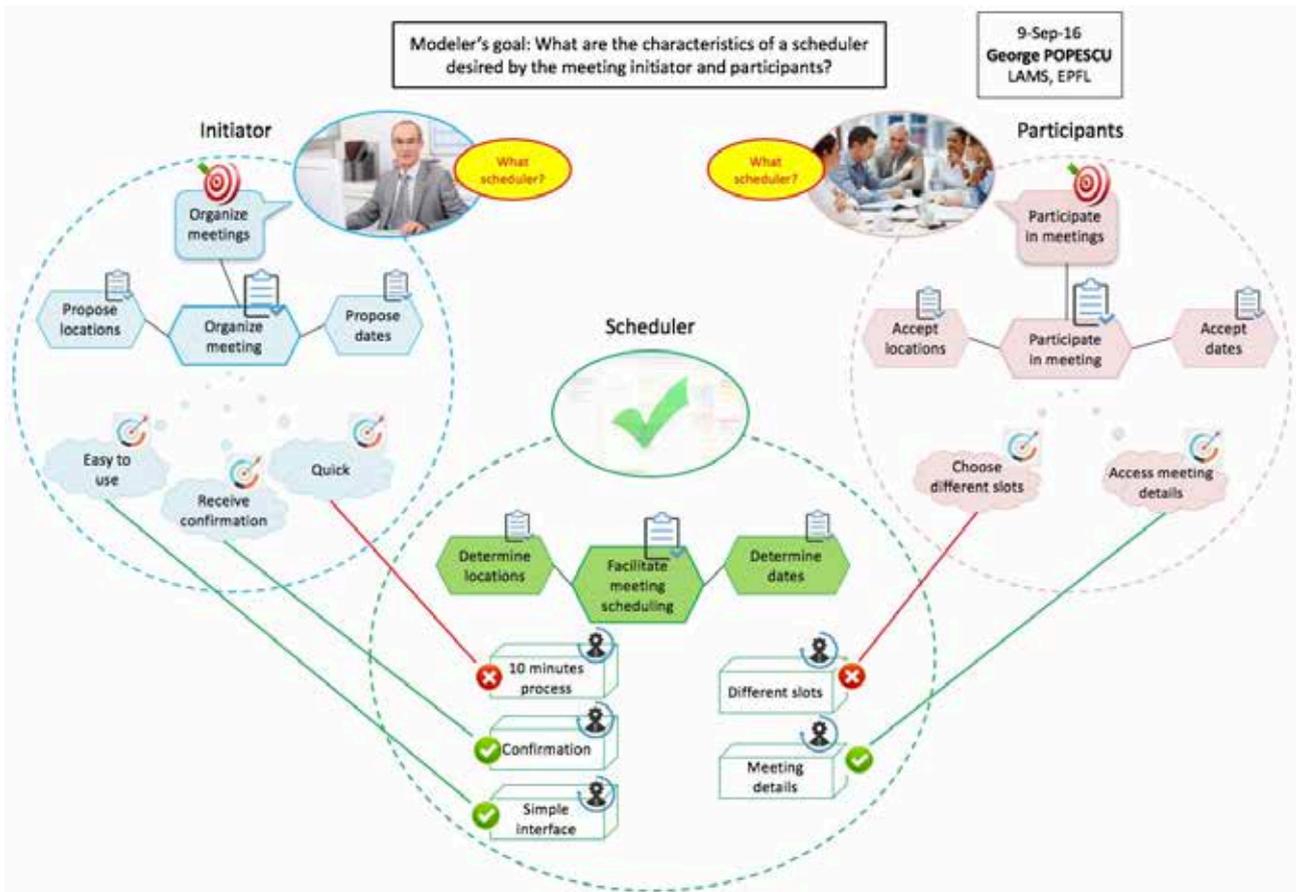


Figure 61: Rationale based on Question, Option and Criteria with the proposed i*

5.5.3 Story

We use the four phases (context, conflict, climax and closure) to create a storyboard model. This model includes a sequence of one to three model instances for each story phase, which helps the modeler convey his story (Figure 62).

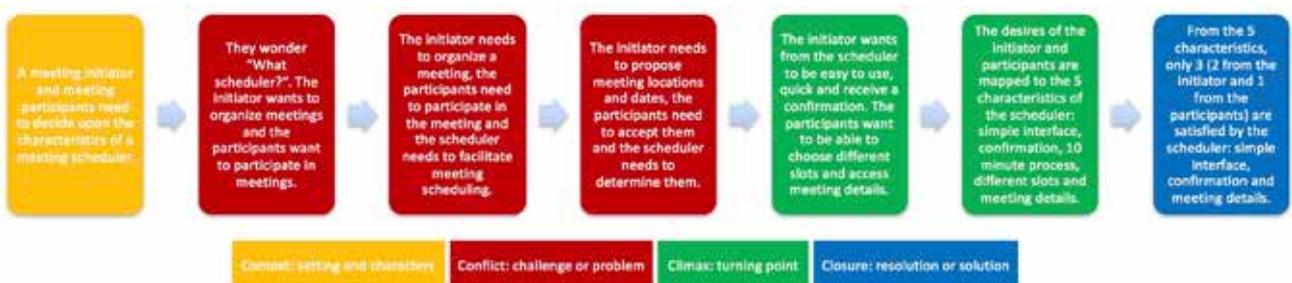


Figure 62: Storyboard design for each i* model instance based on the four story-phases

Similar to SEAM, we create one model instance to model the context and the closure because it suffices to introduce the setting and characters and present the solution to the problem. We create three model instances to model the problem because we want to help the readers understand in detail the rationale leading to the characteristics of the meeting scheduler. We create two model instances to model the climax because we want to show how the softgoals determine the resources of the scheduler. Each of these story phases is described in detail in this section.

In organizing the story, the i^* modeler needs to: categorize objects together (e.g., goals of the actors, characteristics of the scheduler), separate the story in model instances that correspond to each story phase, group elements in model instances, and fill in the gaps between the model instances by using visual cues. The passage of time is not shown because there is no chronological sequence of events. However, the decomposition of the main problem into goals, then tasks, sub-tasks, softgoals and resources determines the search of the characteristics of the meeting scheduler or the answer to the main question.

Context

The context is the first part of the storyboard (in orange in Figure 62). It includes the setting and characters. We design the context by using one model instance.

In the first model instance (Figure 63), we show the three actors: the meeting initiator, the meeting participants, and the meeting scheduler. We color the boundary of the meeting initiator with light blue, of the participants with light red, and of the scheduler with light green. We position them as in the original i* model, without overlaps, and add photos / icons for each.

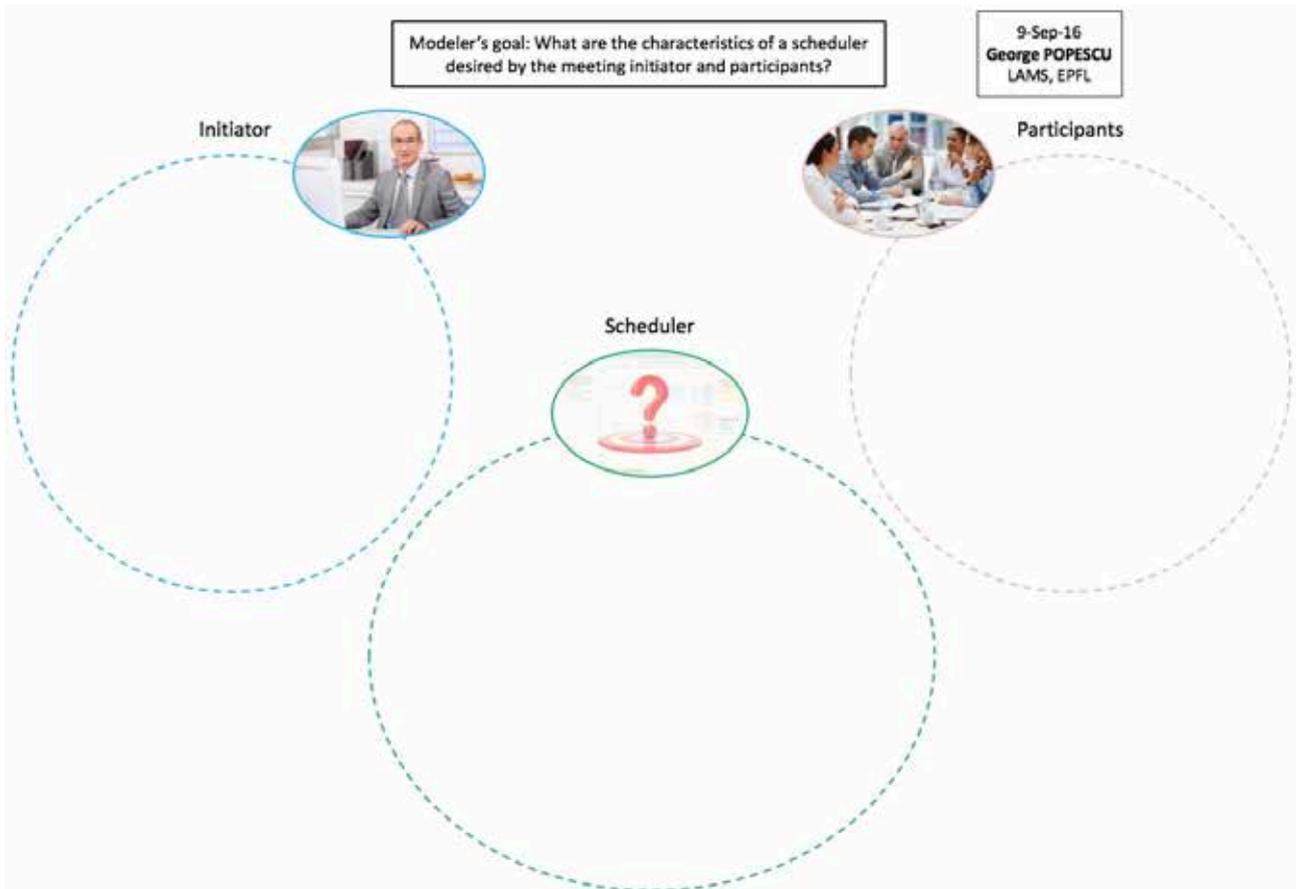


Figure 63: The first model instance of the proposed i* model (context)

Conflict

The conflict is the second part of the storyboard (in red in Figure 62). It describes the challenge or problem. We model the conflict using three model instances. We use the Question, Options and Criteria framework in order to show the rationale more explicitly.

In the second model instance (Figure 64), we introduce the problem by showing it with the red text (“What scheduler?”) on the yellow oval with red boundary. This represents the main question. The option (the meeting scheduler) was previously introduced in the middle. The criteria are subsequently obtained from goals, to tasks, to sub-tasks and to softgoals, and shown in the following model instances. In this model instance, we model the main goal of the initiator (“organize meetings”) and the main goal of the participants (“participate in meetings”). The scheduler does not have a main goal because it represents an IT system.

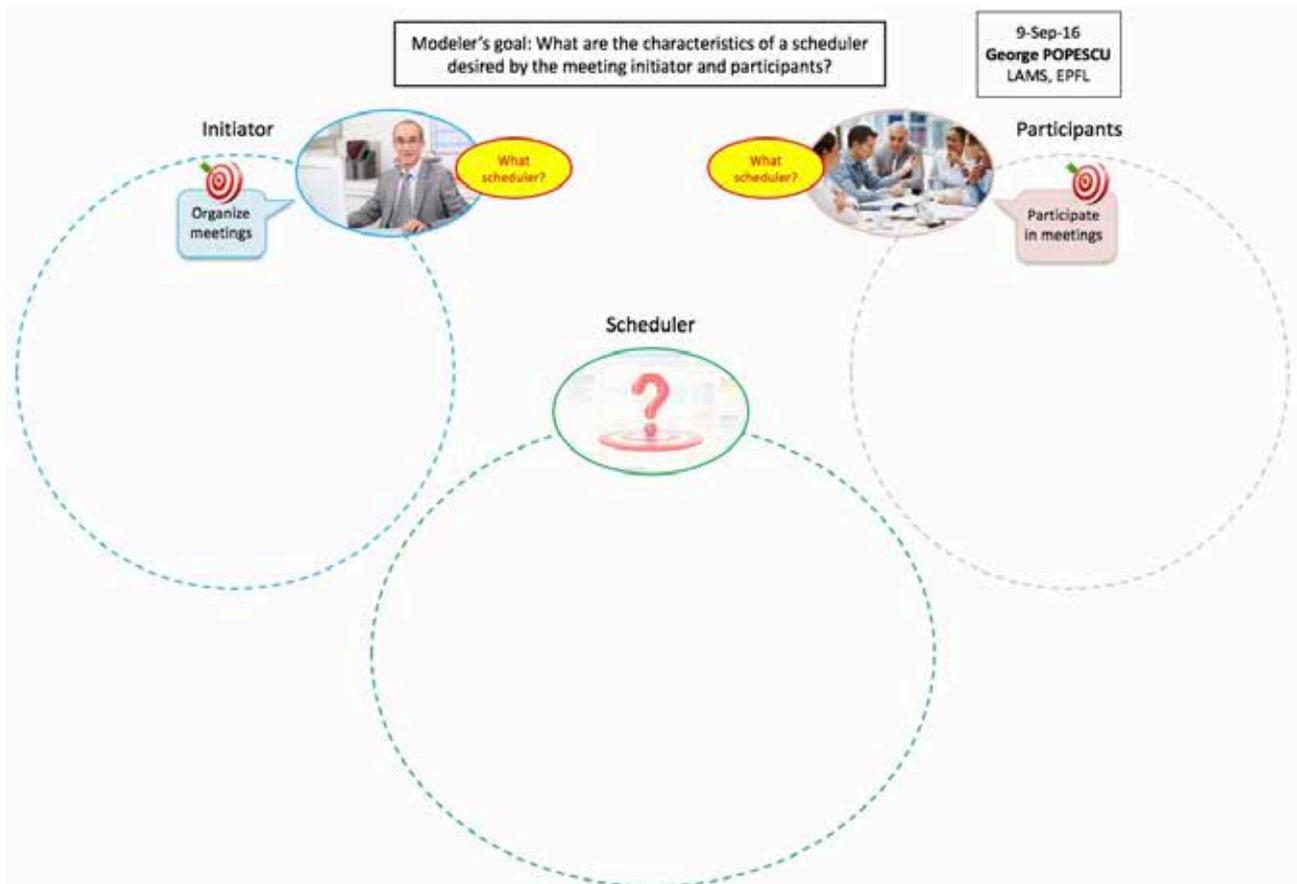


Figure 64: The second model instance of the proposed i* model (conflict)

The third model instance (Figure 65) includes the decomposition of goals into main tasks. For the case of a given meeting, the initiator’s task is “organize meeting”, the participants’ task is “participate in meeting”, and the scheduler’s task is “facilitate meeting scheduling”. For the initiator and the participants, tasks (as actions) are derived from goals.

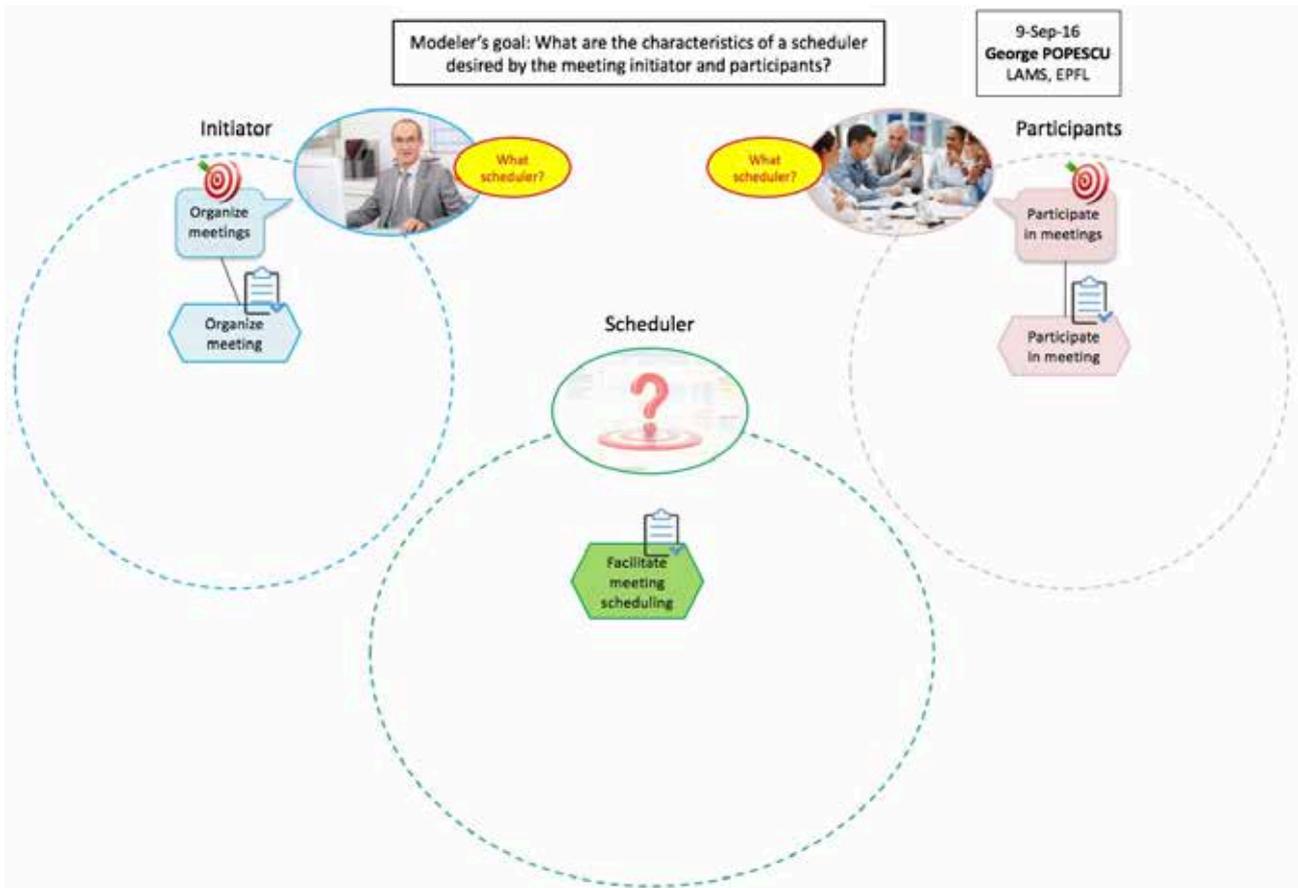


Figure 65: The third model instance of the proposed i* model (conflict)

The fourth model instance (Figure 66) includes the decomposition of tasks into sub-tasks. The sub-tasks of the initiator’s task are “propose locations” and “propose dates”. The sub-tasks of the participants’ task are “accept locations” and “accept dates”. Finally, the sub-tasks of the scheduler’s task are “determine dates” and “determine locations”.

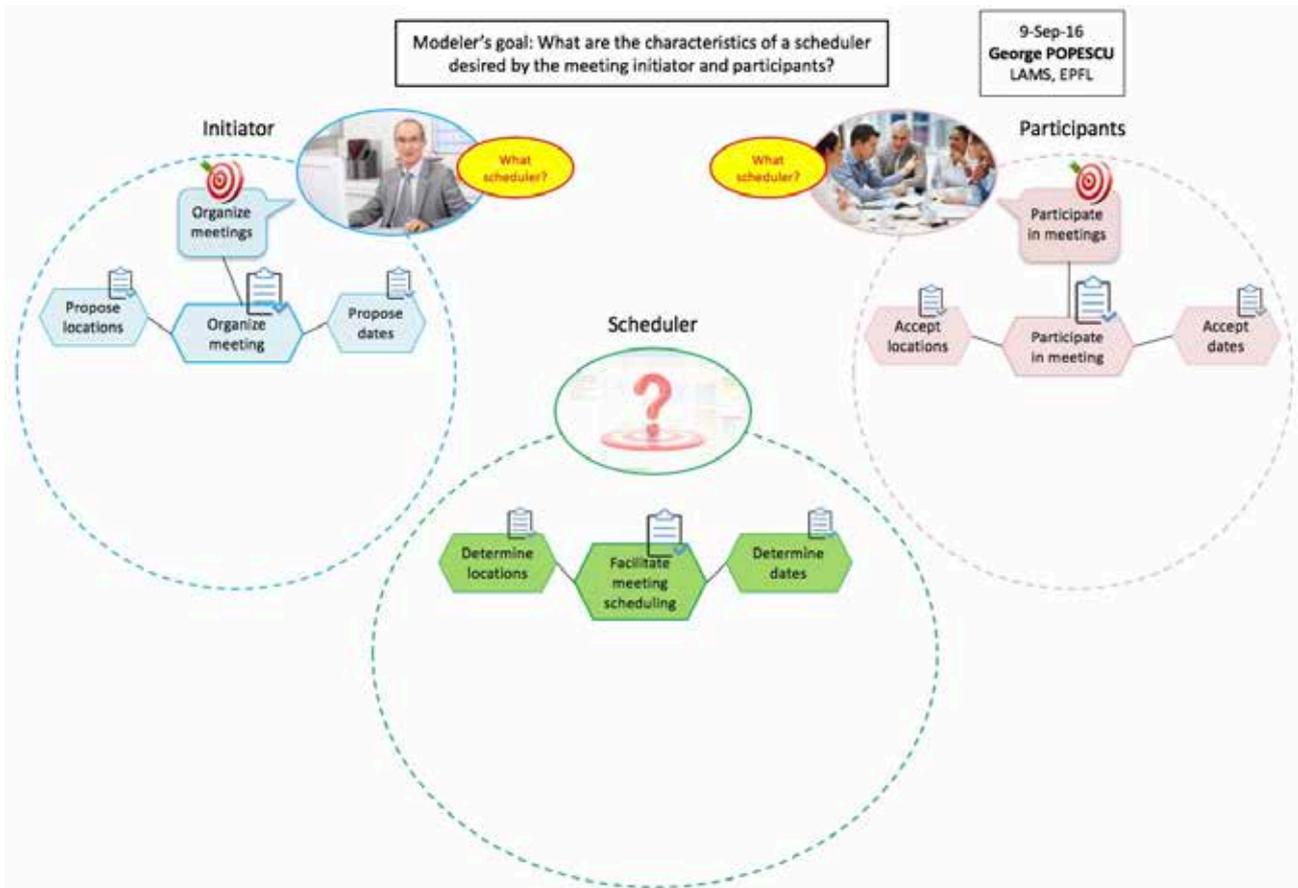


Figure 66: The fourth model instance of the proposed i* model (conflict)

Climax

The climax is the third part of the storyboard (in green in Figure 62). It includes the turning point. We model the climax using two model instances (Figure 62).

The fifth model instance (Figure 67) includes the contribution of softgoals to tasks for the initiator and participants. The initiator has three softgoals: “easy to use”, “receive confirmation” and “quick” that contribute to his main task. The participants have two softgoals: “choose different slots” and “access meeting details” that contribute to their main task. The scheduler, as an IT system, does not exhibit softgoals.

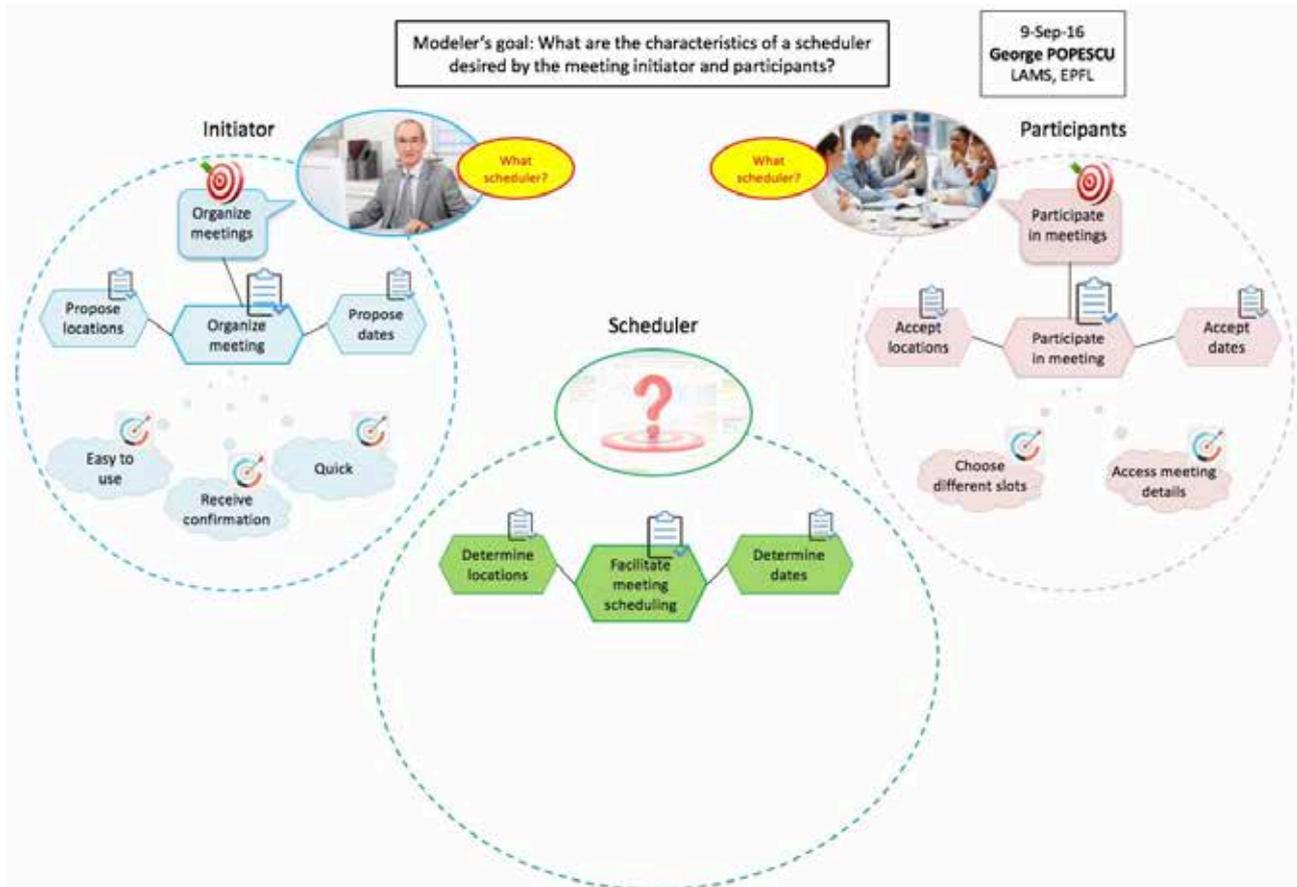


Figure 67: The fifth model instance of the proposed i* model (climax)

The sixth model instance (Figure 68) includes the relations between each human actor's softgoal and the characteristics of the meeting scheduler. These relations are shown using resources. Each of the three softgoals of the meeting initiator and the two softgoals of the participants is mapped to the corresponding characteristic (resource) of the scheduler. For instance, the "Quick" softgoal is mapped to the "10 minutes process" resource and the "Choose different slots" softgoal is mapped to the "Different slots" resource.

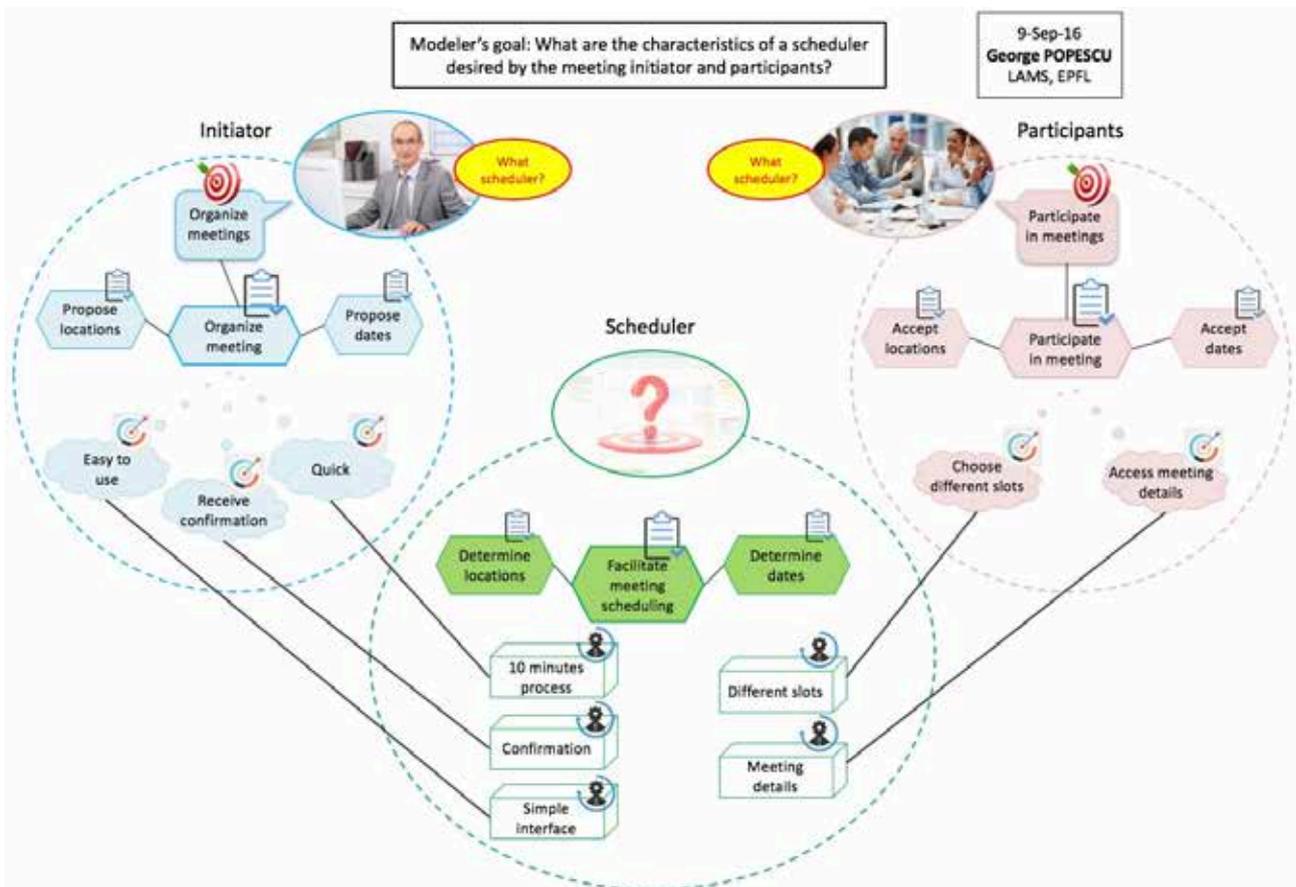


Figure 68: The sixth model instance of the proposed i* model (climax)

Closure

The closure is the fourth and last part of the storyboard (in blue Figure 62). It includes the resolution or the solution to the problem. We model the closure using one model instance.

The seventh model instance (Figure 69) shows the confirmation of the scheduler's characteristics that correspond (or not) to the softgoals of the initiator and participants. From the five characteristics, only three (two from the initiator and one from the participants) are satisfied by the scheduler. These are the "Confirmation", "Simple interface" and "Meeting details".

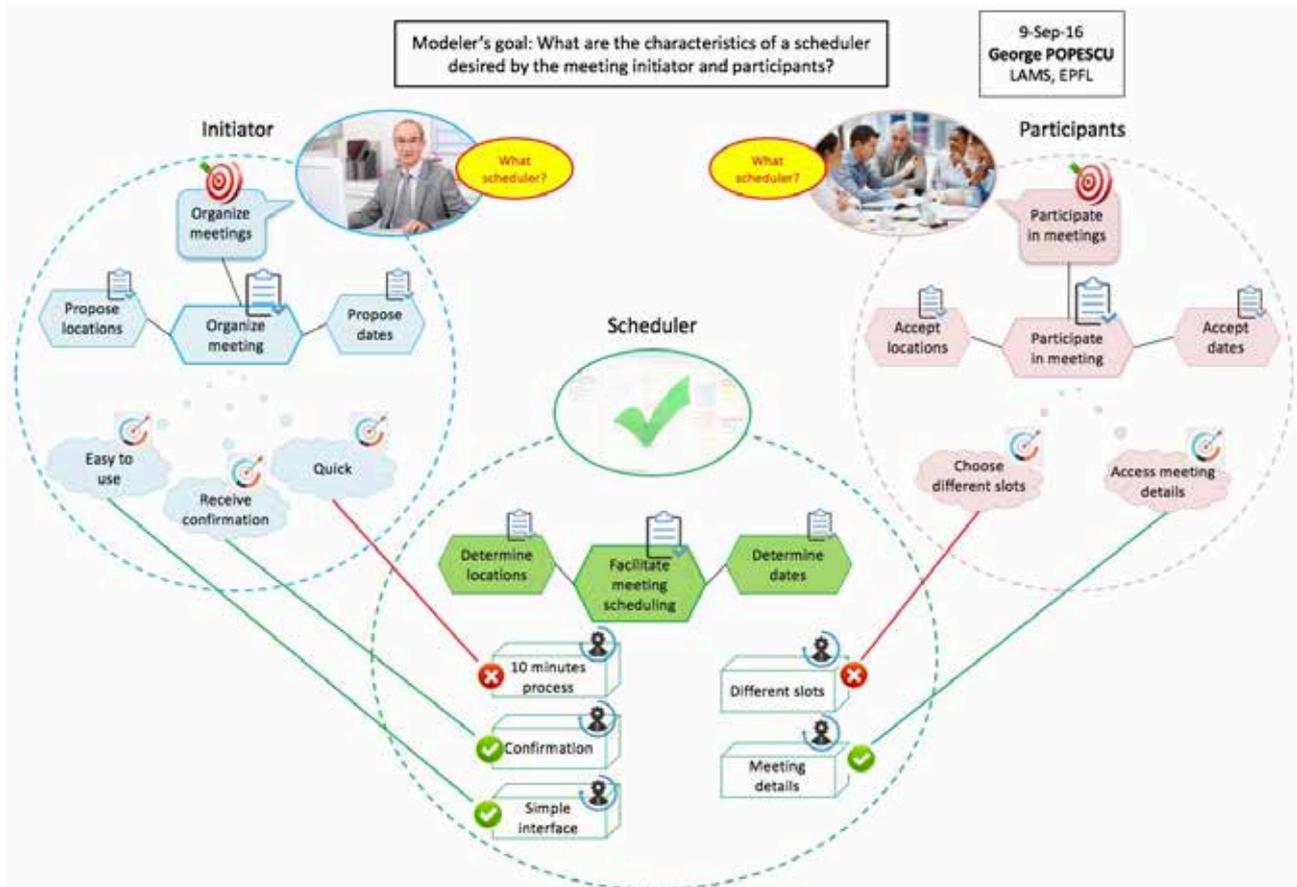


Figure 69: The seventh model instance of the proposed i* model (closure)

Concluding remarks

Similar to the proposed SEAM model, most graphical symbols shown in the proposed i* model maintain the same position. We show gradually the elements that are relevant for each model instance. Except for the last model instance, the meeting scheduler's picture includes a question mark because its characteristics have not yet been specified. When the specification is done, the readers can see which of the characteristics desired by the initiator and the participants are satisfied by the scheduler (correspondence between softgoals and resources).

In contrast with the proposed temporal sequence of SEAM model instances, the i* sequence of model instances does not describe a sequence of moments in time, rather a sequence of increased graphical complexity (i.e., more graphical symbols are shown from one model instance to the next). Actors are shown first, then their goals, then tasks, followed by sub-tasks, softgoals, resources and finally the correspondence between the softgoals and the resources.

5.6 Summary

In Chapter 5, we have presented our research design (Chapter 3.4) applied to evaluate models created with the i* notation. We have shown how to make notation elements more explicit so that readers understand the story. We have created a proposed i* meeting-scheduler model with seven model instances (presented in Chapter 5.5.3). Our contributions to i* modelers are three-fold:

- We model more explicitly the *relation with reality* to show the modeler's goal and model information, actors (human and IT) and their goals, tasks, sub-tasks, softgoals and resources, by using photos, icons and terminology that correspond to readers' conceptualizations
 - Include perceived real-life occurrences of actors (e.g., photos of a meeting initiator and participants) to show concreteness
 - Include visual symbols (e.g., meeting scheduler, goals, tasks, sub-tasks, softgoals, resources) that readers can relate to
- We model the *rationale* (why the problem is meaningful) of actors by adding the main question (oval with yellow background and red boundary), showing the option (scheduler), the criteria (softgoals) and their evaluation (using the check-mark notation for the correspondence between softgoals and resources)
 - Model actors' goals and softgoals that contribute to their rationale
 - Model actors' outcome as a consequence of their rationale (the characteristics of the meeting scheduler)
- We model the *story* by using the context, conflict, climax and closure story-phases to show how the problem (the characteristics of a meeting scheduler emerge, based on the desires of a meeting initiator and meeting participants) is solved (these characteristics are identified: some correspond to the desires, others do not)
 - Show the actors and the main problem that they try to solve, then the decomposition of their goals into tasks, their tasks into sub-tasks, their tasks into softgoals, and the relation between softgoals into resources

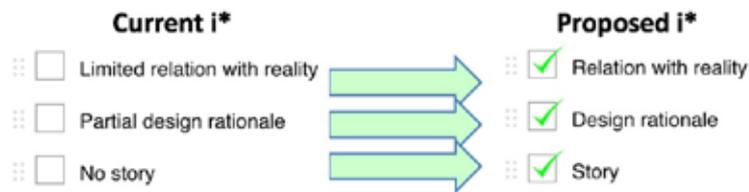


Figure 70: Contributions to i* - from the current i* to the proposed i*

Instead of creating two distinct views of a situation using the i* Strategic Dependency and Strategic Rationale models, we propose one model, constructed using story-phases, that shows one coherent view. The model presented in this chapter represents our best proposal after interviewing twenty readers. It is not the best in absolute terms. The main part of the research work consisted in identifying what is important to be shown in the model and how to make the story clear for readers.

These contributions were validated by readers. The following reactions are from one of the readers who evaluated the proposed model: “The meeting initiator wants to organize meetings based on dates and locations”. “The process has to be easy to use, he has to get a confirmation and has to finish quickly.”, “The participants want to participate in meetings, again based on dates and locations.”, “They need to choose different slots and access meeting details.”, “Now the initiator and the participants want to know what scheduler to choose.”, “The scheduler helps them to choose dates and locations”, “It satisfies only three out of the five desires of the initiator and participants, but they are still ok with their choice.” The reader’s interpretation of the model corresponds to the modeler’s story.

Chapter 6 Theoretical Principles and Contributions

We present the main theoretical principles of the current research, as well as the contributions to modelers and designers of modeling notations. We first discuss each of the three contributions of the thesis, by justifying the need for modeling the relation with reality, the rationale and the story. We indicate the underlying works and theoretical principles for each contribution. Then, we summarize our contributions to modelers of IT modeling notations. Using the contributions to modelers, we present the contributions to designers of IT modeling notations. We present the validation of our recommendations with modelers. We conclude by discussing several limitations of our contributions.

6.1 The Need for Modeling the Relation with Reality

This research is based on the entanglement between the reality, the modeler, the readers and the model. This entanglement originates in the modeler's and readers' conceptualizations of their observed realities (Figure 71). By conceptualization of reality, we mean an explanation of how someone understands what he observes. We extend the conceptualization and modeling framework from (Regev, 2003) by including readers. Our assumption is that modelers need to create models to communicate with readers. These models need to be understood by readers. Therefore, understanding the readers' conceptualizations in relation to the modeler's conceptualization is essential for our research.

In Figure 71, the modeler's conceptualization of the reality he observes is marked with a black background. Black is used to denote the modeler's expert knowledge of using an IT modeling notation (shown to the left). This conceptualization is extensively described in academic and industry papers that detail the use of notations to model various cases and scenarios. However, it is not commonly available for (model) readers. A first segment of readers, such as Reader 1, might have a partial conceptualization about the modeling notation (marked with a black and blue gradient background). For instance, IT-educated readers can relate to the model's notation. They can make analogies with IT notations they are familiar with. A second segment of readers, such as Reader 2, might not be accustomed with the modeling notation and have an entirely different conceptualization (marked with a blue background). For instance, some business-educated readers can infer other meaning than what the modeler wanted to show, based on concepts they are familiar with. A third segment of readers, such as Reader 3, might not even have any conceptualization (marked with a white background). These readers might not be able to relate at all to the concepts presented by the modeler. Readers 2 and 3 do not observe the reality in the same way as the modeler, whereas Reader 1 partly understands what the modeler wants to communicate.

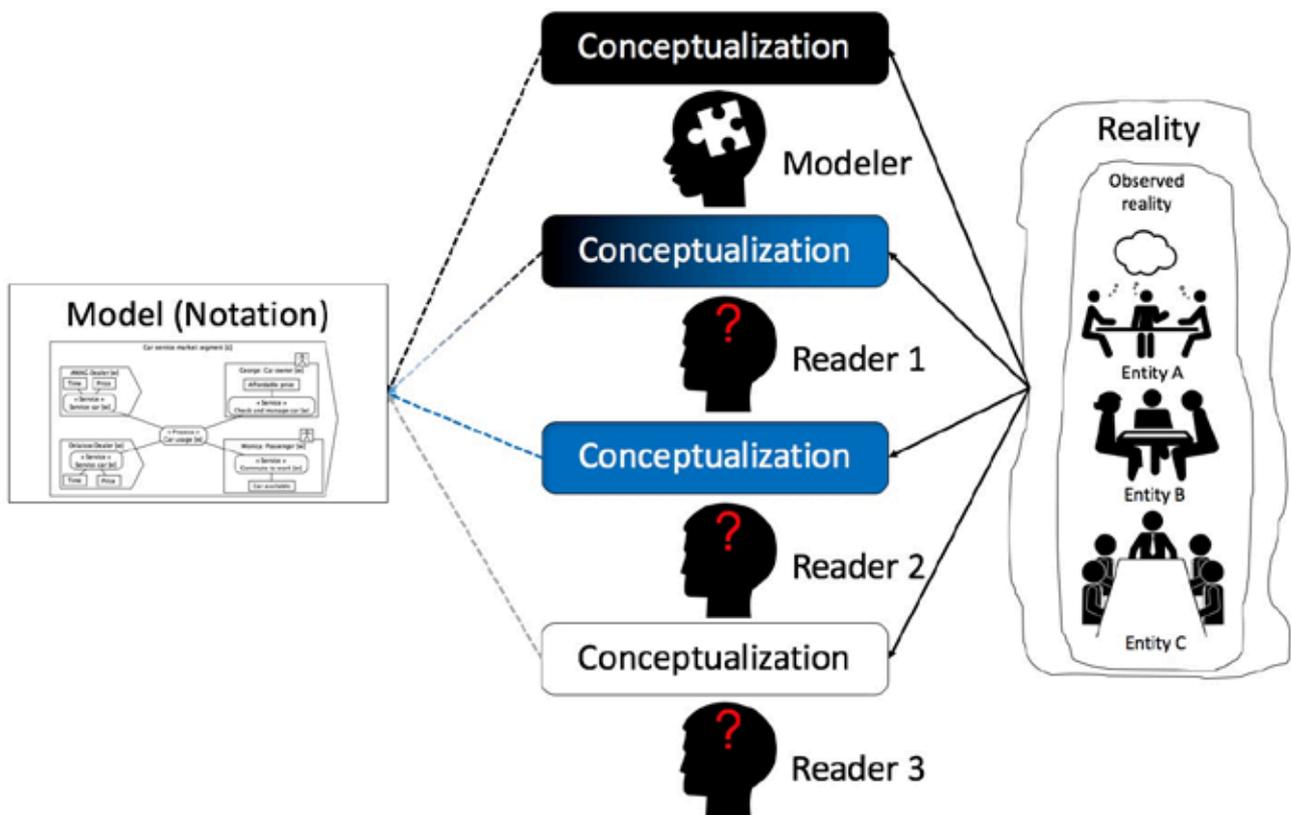


Figure 71: Relations between reality, conceptualization and model – adapted from (Regev, 2003)

Modelers have difficulties teaching readers their conceptualization, especially to readers who have a different conceptualization (Reader 2). They suppose that they have to provide readers with new knowledge to make them understand what modelers observe in the reality. This is challenging because readers already have their own conceptualization (marked in blue) based on their own experiences, hence the problem of adaptation of conceptualization arises. This is explained by the accommodation principle (Weinberg & Weinberg, 1988) that refers to the fact that a compromise between the modeler's and readers' conceptualizations needs to be made. Specifically, the modeler needs to adapt his conceptualization of reality in his models to the conceptualization of his readers in order to enable readers to understand it. His conceptualization needs to be close to the conceptualizations of his readers.

In Figure 71, the modeler's conceptualization of reality is expressed with one of the two modeling notations analyzed in this thesis. The model is created with the original SEAM notation. This conceptualization and modeling framework (Figure 71) can be applied to other models created with other modeling notations. By accommodating the modeler's conceptualization to account for readers' conceptualizations, another conceptualization is formed. This new conceptualization is based on the overlap between the modeler and the readers' conceptualizations (zone of proximal development) and is expressed in the form of the two proposed models presented in this research, one created with the proposed SEAM notation (Chapter 4.5.3) and the other created with the proposed i^* notation (Chapter 5.5.3).

6.1.1 The Identification of the Zone of Proximal Development

We consider that readers discover new meaning and therefore learn from the story that the modeler wants to convey. In his analysis of learning, (Vygotsky, 1997) mentions four stages of the devel-

opment of human knowledge. In the first stage, the empirical discovery is important for the revision of existing views concerning the observed reality. In the second stage, the conceptual form is built on related problems and phenomena from the observed reality. In the third stage, the transformation of the conceptual form into an explanatory principle enables the observer (child) to apply it to any problem in a given discipline. In the fourth stage, the general principle becomes a general methodology applicable to all fields of knowledge. This understanding of learning can be used to explain how readers learn from the models created by modelers. First, readers discover existing views of the observed reality. Second, they form a conceptual understanding of the model. Third, they form explanatory principles that they can apply to other problems and disciplines. Fourth, they develop a general methodology applicable to all fields of their knowledge.

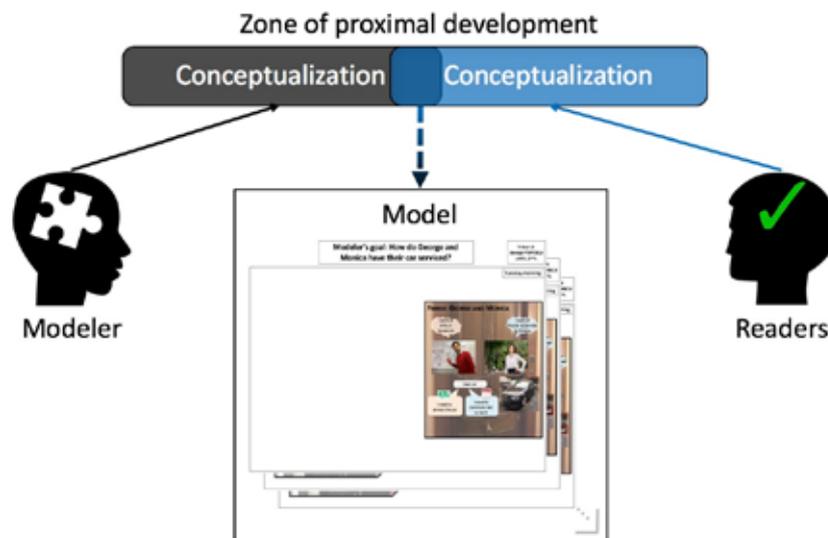


Figure 72: Zone of proximal development between the reader and the modeler

(Vygotsky, 1997) describes the dialogical character of learning. He explains that a child’s disorganized spontaneous concepts “meet” the systematic logic of adult reasoning in the “zone of proximal development” (Figure 72). We use the concept of “zone of proximal development” to explain our results. Despite the fact that the modeler and the readers have different conceptualizations (e.g., IT and business, respectively), the readers’ disorganized spontaneous concepts “meet” the systematic logic of the modeler’s reasoning. Similarly, readers adapt their understanding from something they know (their conceptualization) to something new (the modeler’s conceptualization). This adaptation in understanding occurs at the conceptualization level, not at the level of the observed reality. Graphically, in Figure 71, this means passing from the white or blue backgrounds to the black background. In order for readers to understand what the modeler is conveying, there needs to be at least an overlap between the conceptualizations of the modeler and of the readers. The modeler needs to investigate the conceptualizations of his readers (the way they observe reality), because he does not know or have access to their observed realities. He needs to find “the common denominator” of most readers’ (that form his audience) conceptualizations and create a model based on it. This is the reason that, during workshops, modelers typically start by analyzing elements that are commonly perceivable to most, if not all, readers. As soon as an overlap between the conceptualizations occurs, the modeler can exploit it to teach readers more about his conceptualization. The modeler might rely on knowing his readers’ backgrounds (e.g., readers who have serviced their car or have used a scheduler) in order to use a vocabulary familiar to them, or use common sense. A zone

of proximal development between the readers' and the modeler's conceptualizations is formed (Figure 72). This implies a transfer of knowledge (black background) from the modeler to the readers, allowing the readers to learn. By presenting readers something they know in his model, the modeler extends their understanding. This extension should not be "too tight or too loose". It does not represent an absolute reality with an absolute story, but rather the agreement of perceptions about the modeler's and the readers' perceived realities in a story that fits a purpose.

(Ausubel, Hanesian, & Novak, 1978) describe learning as an active process, rather than a response to the environment. Readers seek meaning in models they see by integrating new knowledge with the knowledge they already have, e.g., through past experiences. They compare what they know against the new knowledge and construct an improved knowledge. (Ausubel, Hanesian, & Novak, 1978) consider an individual's "cognitive structure" as "the sum of all the knowledge that they have acquired together with the relationships among the facts, concepts and principles that form this knowledge". The authors explain meaningful learning as the process of enabling the cognitive structure to discover new elements and enlarge itself. Meaningful learning occurs when new information is digested by the reader's cognitive structure and is connected to the other information that he had previously learned. By opposition, rote learning occurs when the new information is not related to the previously learned information. Therefore, if a reader already has relevant information in his cognitive structure to which the information presented by the modeler in the model can be connected, then his learning can be meaningful. In contrast, if a reader does not already have relevant information in his cognitive structure, then the new information can only be learned in a rote manner, i.e. not constructed based on the model's information. (Ausubel, Hanesian, & Novak, 1978) describes the process of "subsumption", by which new information is brought into a reader's cognitive structure and systematically compared and contrasted with his prior knowledge.

It becomes necessary for the modeler to understand the readers' conceptualizations and for the designers to understand the modeler's and readers' conceptualizations. As explained before, the modeler and readers' conceptualizations need to come together in a zone of proximal development that allows readers to learn from the modeler's conceptualization. The model is therefore a medium or vector for learning. Modelers need to understand the conceptualization of their audience. They need to work with readers to align their conceptualizations, instead of impose their own perceived reality in the model. They should use appropriate photos, icons and terminology that are close to the readers' conceptualizations instead of terms that are specific to the modeling notation and used by expert modelers. Similarly, designers, too, need to understand the conceptualizations of their target audience (modelers and readers).

As designers of modeling notations, we design notations so that modelers of modeling notations can create models that readers understand. Our contribution to modelers is the use of a conceptualization that is as close as possible and overlaps to the readers' conceptualizations, instead of solely the modeler's conceptualization that contains expert knowledge (based on notation-specific terms). We do so through the construction of a zone of proximal development between the modeler and the readers. This zone enables modelers to identify common entities in the readers' observed realities based on their conceptualizations. This justifies the principle of concreteness in modeling the relation with reality. Concreteness is represented in models with elements that suggest meaning (photos, icons and terminology) based on readers' conceptualizations marked by their observed realities.

Figure 73 represents the instantiation of the conceptualization and modeling framework in Figure 71 to the example of the SEAM modeling notation. There are two readers, Reader 1 and Reader 2. Reader 1 might be an IT-educated person familiar with certain elements of the situation shown in the model, based on her experience (e.g., know about the AMAG dealer, George and Monica's car, their place). Reader 2 might be a business-educated person familiar with other elements of the situation shown in the model, based on his experience (e.g., know about the Delaisse dealer, the offers and bills sent by it). Between the two readers, there are commonalities, shown in the overlapping region of their observed realities (such as the fact that both know the two actors George and Monica, the public transportation system in Geneva: tram, and the local currency; CHF).

Our contribution, for both SEAM and i* modelers, is the use of a conceptualization that is as close as possible to the readers' conceptualizations; by using it we establish a relation with reality. By interviewing readers, we learned about how they interpreted concepts and formed ideas (conceptualizations) based on their perceptions of reality. We used the zone of proximal development between the modeler's and the readers' conceptualizations to model elements inspired by the readers' conceptualizations. We included entities from the readers' observed realities in the models we created. Therefore, our models show concreteness in modeling the relation with reality, through the use of appropriate photos, icons and terminology related to model actors.

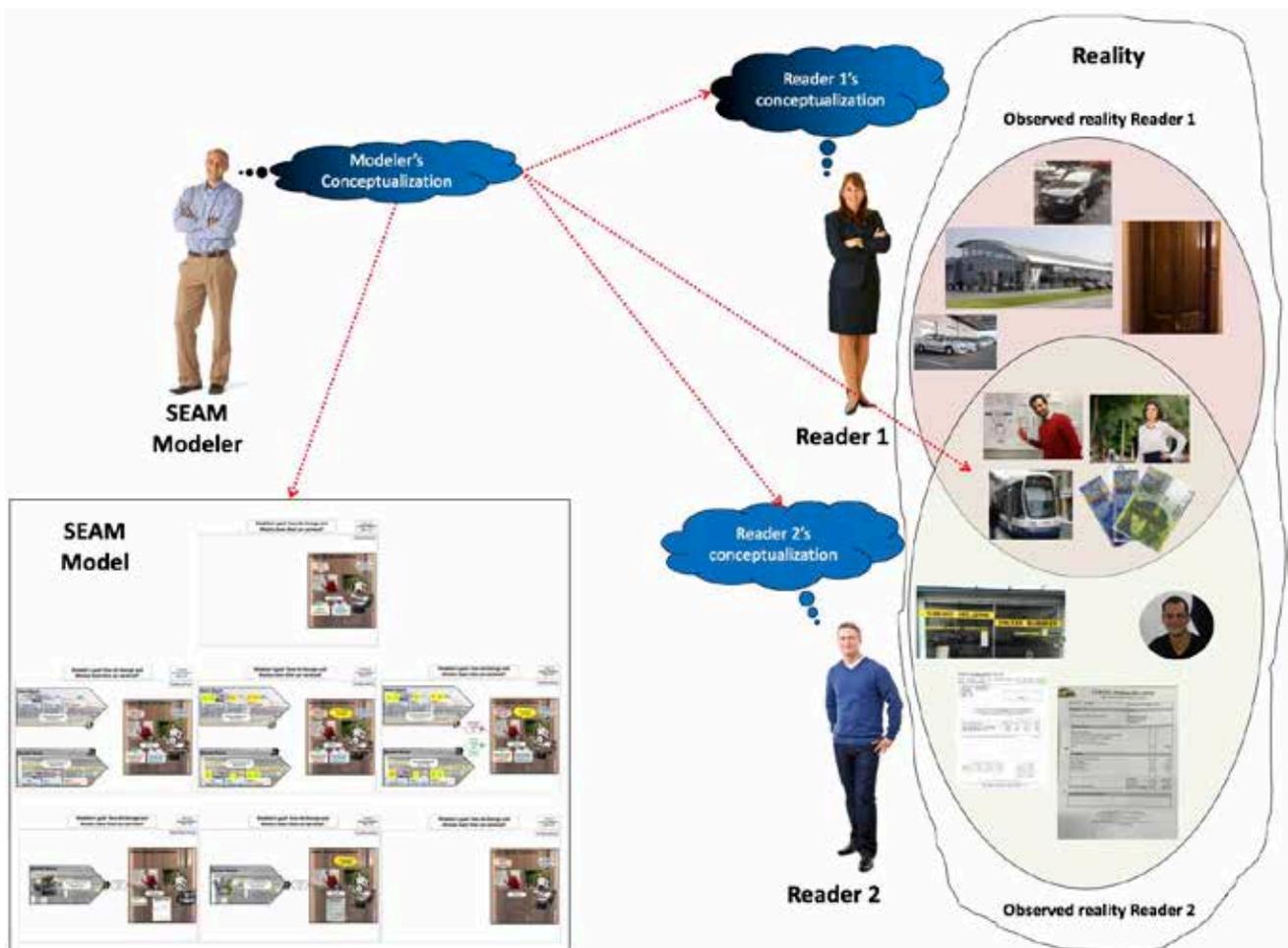


Figure 73: Relations between reality, conceptualization and one model for SEAM

6.1.2 The Alignment of Interpretations

As (Maturana & Varela, 1987) observe, the relation between models and the observed reality takes two extreme views in the mind of the modeler and the readers: solipsism and positivism. Solipsism considers that what exists is defined only by one's inner imagination. The "solipsist" believes that reality does not exist outside his mind. By opposition, positivism considers that each observer applies a filter on reality. The "positivist" believes that an objective reality exists outside his mind. This reality is independent of his interpretation.

Neither of the two views is right as "reality" lies in the relation between the observer and the observed, i.e. between the modeler or readers and their observed reality, respectively. As (Weinberg G. M., 1975) states: "We may believe the world to be independent of the percipient observer, but we definitely feel it depends on the participant observer". Half-way between the two extremes is interpretivism (Maturana & Varela, 1987). The "interpretivist" forms interpretations of the world that are dependent on it and are based on his continuously updated experience. His conceptualization changes continuously based on the observed reality. Therefore, he actively constructs his own perceived reality. The observed reality is what he interprets as his reality. In our conceptualization and modeling framework (Figure 71), both the modeler and the readers are interpretivists. Their interpretations are neither pure fiction nor objective accounts of reality. Modelers should not be surprised if readers misinterpret something in the model that they have not thought of. They should first observe readers' interpretations. Then, they should proceed to improving their models by using improved visual elements that suggest meaning.

Modelers express their own interpretation of reality in models they create by perceiving it in their own personal way. As seen before, readers might form other interpretations that might be different from that of the modeler. In this thesis, we help modelers minimize these misinterpretations. Using the functional notation from (Weinberg & Weinberg, 1988), we explain that the modeler would like readers to observe $z = f(x, y)$ = the story that he wants to show in his model. However, he can only achieve $z = f(x, y, \text{reader})$ = the story that one reader observes in the model. What should be interpreted should correspond as close as possible to what was observed, but readers might interpret the model in unknown ways. Therefore, the reader part of the function should account for only a small, negligible distortion of the model's interpretation (epsilon). The modeler cannot guarantee a complete overlap (identity) between the two functions, nor is this the purpose of his modeling. He should aim for a "good enough" approximation. One necessary condition for achieving this approximation is the alignment of readers' interpretations of the model, on the one hand, and the alignment between the modeler's interpretation of reality and the readers' interpretation of reality, on the other.

Further to the divergence in interpretations of different readers, the "observational problem may be greater when systems of greater complexity are involved" (Weinberg G. M., 1975). Indeed, modelers need not create complex (consisting of many different parts) or complicated (consisting of many interconnections among parts) models, rather simple, evident and easy-to-understand models to convey the modeler's story both effectively (by showing the appropriate elements in the model and by framing a consistent story) and efficiently (by illustrating the story with the least effort in shortest time). When the observer focuses on a particular model aspect, he demonstrates his belief that it is important for his readers. (Weinberg & Weinberg, 1988) describe the "fallacy of incompleteness"

which states that, at some stage of the story, the reader may have omitted some model elements and therefore his understanding of the model will be an “approximation”. This approximation would be good, if the readers still understand the story envisioned by the modeler, and would be poor if they understand a different story, or if they are confused by the one shown in the model.

In our interviews, we noticed that different readers interpret models in different ways. As mentioned by (Ashby, 1961):

“The space may change if the observer changes; and two observers may legitimately use different spaces within which to record the same subset of actual events in some actual thing. The “constraint” is thus a “relation” between observer and observed; the properties of any particular constraint will depend on both the observed and on the observer. A theory of complex systems needs to be concerned with properties that are relational between the observed and the observer.”

The meaning of the model, as perceived by the readers, is based on the relation between the model, the readers and their observed realities (Figure 71). According to (Weinberg G. M., 1975) this meaning is an emergent property of the relation between the observer (readers) and the observed (the model). In this thesis, we improve the notation, by analyzing this relation between the readers and the model. Still, it is the choice of the modeler of what to include in his model, given the story that he wants to show. As mentioned by (Weinberg G. M., 1975) modelers create models so that readers identify “emergent” properties of the model, properties that do not exist in the parts (model elements) but that are found in the whole (either in one model instance or a series of model instances). Properties “emerge” for a particular reader when he cannot predict their appearance - he is surprised when he discovers meaning. There can be cases in which a property is “emergent” to one reader and “predictable” to another.

Each reader develops a unique and valid interpretation of the model, different from that desired by the modeler and expressed with misinterpretations. For our research, the analysis of our models by readers represents useful feedback for identifying elements that are difficult to understand in models. Knowing these elements enabled us, as modelers, to reflect on the possible interpretations of readers, even before making a model decision. Through interviews, we listened to readers’ interpretations and aligned them. In our proposed models, we make difficult-to-understand elements more explicit using visual cues, so that readers’ interpretations converge to the modeler’s interpretation.

In order for modelers to avoid the “fallacies of absolute thought”, (Weinberg G. M., 1975) proposes to “remember the human origins of our models, instruments and techniques”. These human origins refer to the modeler’s conceptualization shown in the model. When modelers create models, by thinking in a certain way, they are “usually following conventional patterns, patterns that will work out well if the situation remains conventional, which most of the time it will” (Weinberg G. M., 1975). Therefore, modelers can make use of these patterns that guide the readers’ thoughts, through visual cues (photos, icons and terminology related to of actors) that reveal concreteness. These patterns are based on our “human origins” (Weinberg G. M., 1975) and can contribute to the alignment between the modeler’s and readers’ interpretations. Some patterns might work against the story that the modeler wants to show.

6.1.3 The Agreement on Boundaries

The purpose of the modeler has to be understood by model readers. Different models can be built for different purposes and for different readers. Modelers need to connect with their readers via the story they show in their models, whereas readers need to interpret the modeler's story through the decisions that modelers faced when modeling. As put by (Weinberg G. M., 1975), when the modeler includes a given relation in the model, or omits it, whether he does well or not, depends on the readers' understanding of the reasoning behind the modeler's decision. Therefore, the modeler needs to delimit his reasoning of the particular situation he models.

The idea of a separation of, or a boundary on, one part from the whole is central in systems thinking (Checkland, 1999). Modelers use the term "system" to mean the "inside" and "environment" to mean "outside". Either can be called "the system", "for one man's system may be another man's environment" (Weinberg G. M., 1975). The choice of a boundary set by the modeler influences the readers' interpretations. Readers are also influenced in their choice of boundary by past experiences, for instance, those determined by physical features of objects or steps of a process. The modeler needs to choose where to stop the process of story description in his model because of the possible excessive amount of detail related to the story.

(Maturana & Varela, 1987) observe that the easy task in model creation is the identification of elements that make up the model. This identification facilitates the construction of a context. The difficult task is to describe exactly and explicitly the relations between elements in the context. Knowing these relations enables readers to create possible interpretations of the evolution of the model elements. Modelers need to consider not only elements in their respective contexts, but also the explicit relations between them. The solution that (Maturana & Varela, 1987) propose is to embrace a broader context. Each observer perceives unity in a set of different domains, depending on the distinctions he makes. On the one hand, a system or organization can be considered in the domain where its elements operate (composite view in SEAM or "inside"). This is the domain of the actors' internal states and their structural changes. Considered as such, the environment is irrelevant. On the other hand, the system or organization can be considered to interact with its environment. This perspective (whole view in SEAM or "outside") enables the observer to establish relations between certain features of the environment and the behavior of the system. As such, the internal dynamics of the system are irrelevant. The example offered by (Maturana & Varela, 1987) is that of a person steering a submarine, he sees only its internal technical machinery. From the outside, an observer perceives its behavior and global properties but cannot see the technical machineries. Therefore, the meaning is given by the choice of perspective.

The modeler decides on the choice of perspective that is meaningful for the story he wants to convey in his models. However, due to the possible relations between the systems he represents, readers might interpret other perspectives and perceive elements in other contexts. By testing our models with readers, we understood that an agreement on boundaries needs to be established. These boundaries are expressed at different levels: the modeler's goal (what the modeler wants to convey with the model), the model (what is needed to be represented in the model) and the model elements (what is needed in order to show certain model elements), in their respective contexts.

(Weinberg & Weinberg, 1988) observe one paradox of perception: "To a certain extent, mental power can compensate for observational weaknesses (e.g., attention span). To a certain extent, ob-

servational power can compensate for mental weakness (e.g., memory span)". In this work, we show how mental weakness can be compensated with observational power by reducing the misalignment between the modeler's perception of the model and the readers' perceptions. Furthermore, even a "super-observer" cannot remember all the scenarios and represent all their possible variations. Readers are omniscient (all seeing) but not omnipotent (all powerful) as their attention span is limited. As (Weinberg & Weinberg, 1988) notice, "super-observers do not have power at all: they are omniscient and impotent at the same time". This observational problem can increase when systems of greater complexity are shown in the model.

The approach presented in this thesis helps the modeler focus on certain elements by using visual cues. These elements guide readers' thoughts towards the modeler's interpretation of the situation. When readers make sense of a visual cue that corresponds to what the modeler wants to convey, it means that the model is well suited for readers. When readers find difficult to interpret visual cues or the relation between them, it means that the model requires too much effort and that the visual cues were not sufficiently explicit. When readers misinterpret the context in which the model elements operate it means that the boundary was not well defined by the modeler. Our interviews taught us that readers seek to understand objects in their contexts (defined boundary) in order to interpret the relations between them. Modelers need to explicitly specify the modeler's goal, the model, and the model elements boundaries.

In Chapter 6.1, we have presented our first contribution: modeling the relation with reality. We have justified the importance of the identification of the zone of proximal development, the alignment of interpretations, and the agreement on boundaries. These are based on the understanding of readers' conceptualizations and are useful for reducing readers' possible (mis)interpretations, and for understanding how modelers ground their modeling.

6.2 The Need for Modeling the Rationale

There are numerous problems that modelers address in their models: clarify requirements, communicate to external audiences, communicate arguments to designers, present stakeholders' viewpoints. These are all examples of what (Rittel & Webber, 1973) name "wicked problems". This type of problems describes an insufficiently understood situation that has no right or wrong solution, has no objective metrics to measure success, and requires complex judgements relying on moral, political or professional considerations that cannot be formalized.

Wicked problems can be approached by using argumentation. (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997) offer a description of the Issue-Based Information System (IBIS) method, developed by (Rittel & Webber, 1973), in order to encourage model stakeholders to discuss "Issues, Positions in response to Issues, and Arguments to support or object Positions". In this thesis, the modeling problems that were analyzed, car-maintenance service and meeting scheduler, are wicked problems that are tackled graphically using an argumentation-based rationale inspired by IBIS, named Questions, Options and Criteria (QOC), that is presented below. For both cases, the argumentation was necessary to arrive at a solution. The underlying assumption of the thesis is that if the readers are presented with the reasoning, then they will understand the decision (choice) based on it.

As mentioned in Chapter 2.3.1, the design rationale considers justifications for reasoning about a specific design. (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997) present several efforts for capturing the design rationale by using “graphical argumentation structures and cognitive tools”. They propose the introduction of such argumentation structures into model designs that prove useful for system development, particularly for large scale interactive systems. We believe that argumentation tools can be more broadly used in modeling. Specifically, they can be used for the design of IT and business organizations. As an argumentation tool, the design rationale is useful for providing arguments for the model design, as well as for actors’ arguments within the model. For instance, it is useful to relate the service offered with its implementation and know how to align a system implementation with the design goals (Bajić-Bizumić & Wegmann, 2015). Modelers need to investigate the design-space analysis and make the design rationale more explicit in the models they create, so that readers understand why the model was created and the purpose of the actors shown in models (the problem they try to solve).

One argumentation structure for Design Rationale (DR) is the QuestMap argumentation tool (Corporate Memory Systems, 1993) based on the Questions, Ideas and Arguments (QIA) framework used to “visualize discussions, track unresolved issues, and qualitatively assess the strengths and weaknesses of different positions” (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997). Another one, inspired by QIA, is the Questions, Options and Criteria (QOC) framework (MacLean, Bellotti, & Shum, 1993). The specificity of QOC is that it was designed to represent the design space around artifacts.

As mentioned in Chapter 2.3.1, modelers need to use the Design-Space Analysis (DSA) to discover key Questions, to explore related Options and justify these Options through Criteria and their Assessments of Options (QOC). The design space is not revealed to readers, but the way Questions, Options and Criteria are linked need to be explicit so that the answer to the question is clear. We conclude that DSA and QOC offer an explicit model for understanding the DR. The design rationale can be applied at two levels: (1) the notation level between the modeler and the readers (Figure 74), and (2) the model level between the model actors. Wicked problems cannot be solved using solely the design rationale. The graphical solution needs to consider the “satisficing” (satisfy + suffice) of options with respect to criteria (options that are only good enough instead of satisfying all criteria) and the “accommodation” of conflicting interests in decision situations that need a trade-off between criteria. We explain both concepts in the next sub-sections.

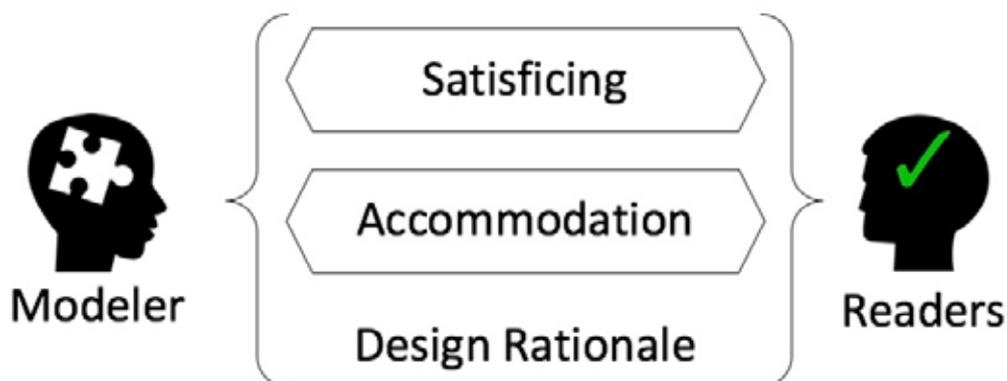


Figure 74: The Design Rationale between the modeler and the readers

6.2.1 The Search for Satisficing Options

People's choices are not always rational and cannot be explained rationally. As (Simon, 1969) pointed out, the concept of "satisficing" is often more important in understanding decision making than rationale. Humans do not make (fully "rational") choices by using algorithms for sorting lists of preferences. Rather, they have needs and express desires for what they choose. Their needs and desires cannot be separated. The choice is sometimes "irrational", constructed on desires rather than needs. As (Gause & Weinberg, 1989) remark, "by clarifying their desires, people sometimes clarify what they really need and don't need". Therefore, a satisficing option is one that partly satisfies criteria and is still sufficient to make a decision (e.g., choice).

In our example, satisficing can occur at two levels: the first, notation level, between the modeler and the readers (Figure 74), and the second, model level, between model actors (e.g., George and Monica for SEAM, and the meeting initiator and meeting participants for i^*). At the first level, the modeler needs to identify visual elements that are "satisficing" for readers – elements that they understand sufficiently well – in regards to the decisions he took when modeling. At the second level, for instance, George and Monica need to identify one dealer that "satisfices" their need to service the car. It might have well been the case that AMAG had a faster and less expensive service offering than Delaisse, and that George and Monica still chose Delaisse. A possible explanation of this "irrational" decision might be influenced by a personal experience of the two customers, such as the fact that George knows well and trusts the Delaisse dealer (this situation is not modeled).

As we learned from readers, they favor the fact that the criteria leading to a decision are rational. They need to deconstruct the decision based on these criteria in a logical way. Specifically, for the example of SEAM, they understand how one dealer was chosen if the evaluation of the criteria leading to this choice is rational. Otherwise, this decision would seem irrational (unexplainable). Design thinking helps modelers understand wicked problems based on satisficing choices – choices that satisfy criteria and are good enough, instead of being evaluated using an algorithm that combines the assessment of criteria.

6.2.2 The Accommodation of Needs

(Checkland & Holwell, 1998) define "accommodations" as conflicting interests that appear in situations that do not fully satisfy every stakeholder but are sufficient to enable actions. Consensus represents a particular case of reaching accommodations. The accommodation applied to the rationale refers to the fact that a compromise between needs and desires has to be made. Accommodation can explain "non-rational" decisions, such as the choice of an option over the other in spite of a less favorable assessment of criteria.

Similar to satisficing, in our models, accommodation can be applied at two levels: the first, between the modeler and the readers (Figure 74), and the second, between model actors (e.g., in the SEAM model, George and Monica need to reach a compromise and trade their preferences with respect to time and money, whereas in the i^* model, the meeting initiator and the participants need to accept a meeting scheduler that does not fully meet all their criteria).

At the first level, the notation, an accommodation appears when not all elements describing a model can be included by the modeler. He has to make certain choices. A tradeoff between spatial (number

of elements included in the model instance) and temporal (number of elements shown gradually between model instances) complexity needs to be made. Modelers, therefore, should find a compromise solution that is not necessarily perfect in the sense of fulfilling all design criteria, due to the constraints related to the perceptions of readers.

At the second level, the model, for the car-maintenance service-case, an accommodation appears when the two actors, George and Monica, need to make a compromise between their criteria. George is concerned about money whereas Monica is about time. The amount of time lost has to account for the money paid. The two actors need to choose a solution that will not fully accommodate both of their concerns: the monetary expense and the lost time. The accommodation can explain the choice of a dealer over another, in spite of less favorable evaluation of the service offering (criteria). Similarly, for the meeting-scheduler case, an accommodation exists when the meeting initiator and participants need to trade their preferences for the features of the meeting scheduler. The two actors need to content themselves with the scheduler that satisfies only three out of five criteria (two out of three for the initiator and one out of two for the participants). The accommodation can explain the choice of a meeting scheduler that does not satisfy all of the initiator and participants' preferences.

In Chapter 6.2, we have presented our second contribution: modeling the rationale. We have explained the main constituents that inspired the graphical expression of the rationale in the thesis models. We have explained how one form of the Design Rationale (DR), namely the Questions, Options and Criteria (QOC) framework, is used to solve wicked problems that accept solutions based on two concepts: satisficing of options and accommodation of conflicting interests.

6.3 The Need for Modeling the Story

Story-telling is the act of conveying a story in a model by revealing elements that encourage the readers' imagination. It relies on human condition (e.g., stereotypes). It involves the interaction between the modeler (as story teller) and the readers (as story readers). According to (Callahan, 2016), when communicating in business and IT, modelers need to rely one part on story-telling and three parts on reason, argument and logic in order to get their message across and inspire action. We explained the reason, argument and logic using the rationale in the previous chapter. In this chapter, we explain the need for modeling the story. For helping people become good business story-tellers, (Callahan, 2016) provides a detailed process.

Modelers know more than they can tell, and they can tell more than they can model. Therefore, they have to make choices of what to model. The story they tell has to be shared in a way that has impact for model readers. In terms of subject chosen, stories can range from personal happenings to experiments and lessons they learned. These stories can be used to make a point about the entire organization that modelers are a member of. When creating story-based models, modelers should emphasize the need for others to find and share their own stories (e.g., tell short anecdotes during workshops). This is because work gets done in organizations because of the people who get inspired to contribute.

(Callahan, 2016) cites a study from 2012 with 300'000 employees, in which more than half did not understand their organization's strategy. One major cause is that this strategy was not effectively communicated. According to (Callahan, 2016), one impactful way to communicate business strate-

gy in organizations is through the creation of succinct story-based models rather than wordy fact-based PowerPoint slides. SEAM can be used to communicate business strategy. Two examples of business strategy communication using SEAM models are presented in Appendix 3 and Appendix 4. (Callahan, 2016) observes that enabling employees to be involved in the model creation is more effective than e-mailing a deck of slides created by a professional design; but it also demands for more effort.

The crafting of a good business-strategy story needs to involve people to develop the story so that they would make the story their reality (e.g., participants at a workshop). Such a story has to enable employees (as readers) to relate to events in their environment. It demands for participants to share their own experiences openly. It also demands for others in the organization to tell their own perceptions of the story so that they collaborate towards building the reality that was told with the model's story. Therefore, stories are not only communicated to readers, but also shared by them.

Stories are simulators for the readers' minds. There are multiple dimensions that story-tellers should consider. First, the story needs to have structure that readers can follow. Second, it needs to evoke emotions cultivating readers' inner reactions enabling them to take action according to the message of the model (e.g., new business strategy). Third, it needs to convey empathy – modelers need to share the emotional response of their model actors and that readers can mirror. Fourth, it needs to be based on the modeler's style of choice and representation. Fifth, the story needs to enable interpretations for readers. Sixth, it needs to be concrete and specific, so that model readers would take action based on the example offered in the model. Seventh, the story needs to refer to a situation in which the readers experience something new, remarkable or memorable, which will enable them to learn something useful. In creating our thesis models, we considered the seven story dimensions.

For the design of a story, (McCabe & Peterson, 1984) mention that a good story provides readers with an orientation about whom and what the experience to be described involves, and about when and where it takes place (context). The story needs to include high points developed through the recapitulation of events (conflict) and then to suspend the action at a crisis point by emphasizing its importance (climax). Then, the resolution of this crisis (closure) is provided through summarizing the events. In the models developed in this thesis, we use the four story-phases (context, conflict, climax and closure) to show how the crisis (problem) is solved and to explain the changes. We constructed between one and three model instances for each story phase. We made the story more understandable to readers by improving the model's structure and content for each model instance.

(Weinberg G. M., 1975) observed that human understanding of systems (perceived entities, phenomena in models) concentrates on two topics:

- “being” is an indefinite moment in time, together with the aspects of the structure that denote the components and their relations that constitute a particular unity and that make its organization real and appear relatively unchanged to an observer; it is shown using “sets, diagrams of structure, properties, boundaries and the white box”. The white box is the implementation of a system or a model of how it operates, as imagined by the modeler.
- “behaving” is a longitudinal moment in time, together with the changes that constitute the functioning of a system that a modeler describes as actions in relation to a certain environment; it is shown using “state spaces, chronological graphs, inputs, randomness and the

black box”. The black-box is the perceived behavior of a system or an abstraction of the white box, as imagined by the modeler.

Modelers should understand the relation between the “being” (white box) and the “behaving” (black box): how a particular behavior leads to the inference of a particular structure via properties and how a particular structure leads to the production of a particular behavior through programs. Furthermore, the “being” determines the “behaving”. “Behaving”, because it represents a sequence of occurrences in time, is easier for modelers to model and for readers to understand.

The thesis is based on the shift “from a focus on organization or structure to a focus on action (behavior), from being to behaving, from form to function, from pattern to process, from timeless to the temporal” (Weinberg G. M., 1975), inspired by readers’ interest in stories. The models developed in this research include systems that present an evolution of actors and objects that show a structure in space to their behavior that shows a structure in time. Novice readers prefer story-based modeling (“behaving”) while expert modelers prefer structure-based modeling (“being”). It is therefore necessary that the modeler compensates the use of structure with the presentation of a story.

From our interviews, we learned that readers always perceive an incomplete view of a model. The model is made out of systems. Systems are interconnected components working together as a whole, a human perception of reality, a viewpoint. Furthermore, they perceive model entities from an additional perspective: “believing” (Weinberg G. M., 1975). Ultimately, readers are entangled with what they observe in ways that leave being, behaving and believing undeterminable.

In order to represent the “being” and the “behaving”, and to reduce the “believing” into one story, modelers need to investigate the states of actors. In order to explain the evolution of the story to readers, they need to represent these states in each model instance, crafting “good stories”. In the following sections we present several story modeling processes.

6.3.1 The Problematic Situation Modeling Process

Soft Systems Methodology (SSM) (Checkland, 1990) analyses a story of a problematic situation with the following dimensions:

- *Customers* - Who are the beneficiaries of the highest level business process, and how does the problem affect them?
- *Actors* - Who is involved in the situation, who is involved in implementing solutions, and what is impacting their success?
- *Transformation process* - What is the transformation that lies at the heart of the system?
- *World view* - What is the big picture and what are the wider consequences of the problem?
- *Owner* - Who owns the process or situation being investigated, and what role they play in the solution?
- *Environmental constraints* - What are the constraints and limitations that affect the solution and its success?

The SSM modeler considers the following six-steps process for modeling a problematic situation:

1. Express the problem situation
2. Define relevant systems

3. Describe conceptual models of systems
4. Compare the model with the real world
5. Propose changes that are systematically desirable and culturally feasible
6. Improve the problem situation

Steps 1 and 4-6 are related to the real world, as the modeler perceives it. Steps 2-3 are related to the modeler's conceptualization about the real-world (e.g., systemic conceptualization for SEAM modelers).

6.3.2 The Comics Modeling Process

(McCloud, 1994) presents a six-step process for modeling stories enhanced with graphical symbols (Figure 75). The modeler (storyteller) begins with an idea that gives him the content of the story (emotions, philosophies and purposes). Then, he explores the form the story will take: text, audio, graphical model (e.g., book, song, sculpture). Third, he uses idioms that illustrate the model's vocabulary and style that are useful to identify the model's genre. By putting all elements (entities represented with graphical symbols) together, the modeler gives his model a structure; he reflects on what to include in the model / what to leave out as well as how to compose his work (arrange it). Then, he crafts the story by applying his skills, practical knowledge and problem-solving skills. Finally, he gives his model an improved surface by identifying the aspects that are most apparent on a first, superficial exposure to readers.

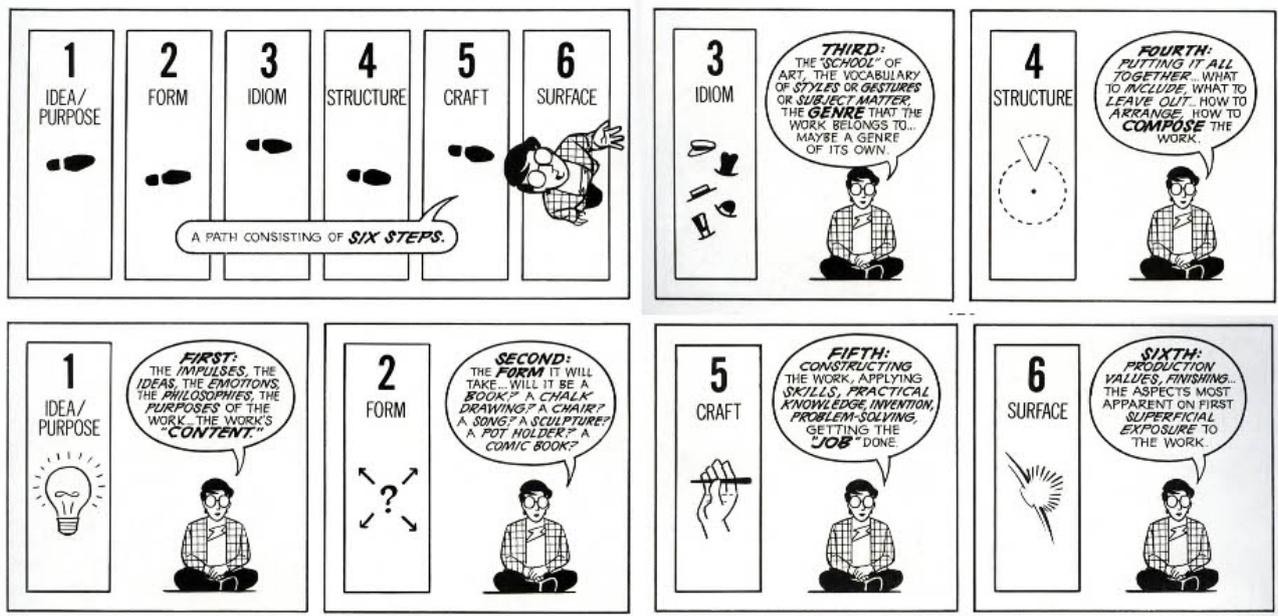


Figure 75: 6-step process for story design – source: (McCloud, 1994)

6.3.3 The Story Design Process

(Brooks & Quesenbery, 2010) observe that “stories revolve around an action done by persons to achieve results”. Therefore, designing meaningful stories that carry an impact is key. The authors teach how to communicate, explore, persuade and inspire readers by crafting unique stories. As technically-oriented modelers are inclined to show facts and figures, the stories they show lack compellingness because they do not allow readers to self-relate with actors shown in the models.

(Brooks & Quesenbery, 2010) propose to modelers to allocate time for thinking about the ethics of their stories (e.g., the moral principles that govern an actor's behavior, the choices he makes), much more than including actions and numbers graphically. In terms of structure, the authors recommend comparing an audio story with a visual one, or moving from a textual model to a graphical one.

6.3.4 The Animation Design Process

RSA Animate represents a way of illustrating and sharing ideas (Park, 2016). RSA Animate is used by modelers to show stories using videos. These videos are rich in visual metaphors (e.g., emotions, highlights). Modelers are also narrators. They introduce model elements by drawing them in real time and explain the advancements of the story orally. The drawing is shown accelerated to match the speed of the narrator's voice. A particularity of RSA-Animate models is that they follow a linear, unidirectional storyline. The modeler zooms in and out on a particular idea to show particularities of it. The multitude of styles and structures of RSA-Animate models enables readers to see the details and maintain an overview of the entire story. The style practiced by every RSA-Animate modeler makes each model unique. The narrator's story uses common sense words that can be followed by readers.

The models presented in this research are different from RSA-Animate models. First, in this thesis models, we do not include the modeler's narration (e.g., using audio). Second, the granularity of our models is determined by the model phases and is represented with model instances whereas RSA-Animate models show a continuous story, without "interruptions" or divisions in instances. Third, our models are a sequence of model instances (frames), whereas RSA Animate models are (fluent) videos (with no frames). Due to these three main differences, RSA Animate models can be considered more explicit and therefore more understandable by readers. However, they also require the necessary skills of modelers for combining drawing, oral communication and video recording with story-telling. An approach, similar to RSA Animate, of a SEAM modeler used to create and explain models is included in Appendix 3 together with the textual extract of his oral narrative.

6.3.5 The Narrative Storyboards Design Process

(Greenberg, Carpendale, Marquardt, & Buxton, 2012) provide sketching methods that modelers can use to achieve good model designs. When sketching, modelers focus on actions and interactions of actors that unfold over time. (Greenberg, Carpendale, Marquardt, & Buxton, 2012) provide a workbook, i.e. a series of instructions and exercises that enable modelers to cultivate a "culture of experience-based design". Most importantly, for the purpose of our research, the authors describe the creation of visual narratives based on "snapshots in time". These allow for the construction of sequential storyboards (capture key ideas in a sequence of frames over time), state transition diagrams (includes interaction states and transitions triggered by interactions yielding in multiple decision paths), branch storyboards (illustrate decision paths over time) and narrative storyboards (present a story about the interaction context, physical environment, actions of people, and events over time).

The narrative storyboards are closest to the models created in this thesis. The main difference consists in the level of zoom or focus that enriches narrative storyboards that was not altered in the thesis models. The background (scene) was kept static / fixed in order to enable readers to identify the elements that appeared / disappeared from one model instance to the next as well as those that moved from one context to another. The main blocks or main actors were kept in the same position

(e.g., the two dealers and the customers for SEAM and the meeting initiator and participants for i*) whereas the elements that provided the conflict and climax were shown successively (e.g., the car moving from the customer side to the supplier side, the display of tasks derived from goals and then the sub-tasks).

The design of stories based on the four story-phases (context, conflict, climax and closure in Chapter 2.3.2) applied to model instances helped readers interpret stories better. For each model instance the modeler should guide the readers' thought. An alternative to the four phases is the structure of narrative storyboards (Greenberg, Carpendale, Marquardt, & Buxton, 2012):

1. Where does the interaction takes place?
2. What is the problem?
3. What is the task that people are trying to do?
4. Which people are present and what are their actions?
5. What kind of objects do they use?
6. What is the possible input / output for each object?
7. How do the actions of people and / or objects solve the problem?

6.3.6 The Comparison of SEAM and i* Storyboards

The three types of current SEAM models (Goal-Belief, Behavior and Supplier-Adopter-Relationship) can be used to represent only two instances of a story, composed of the current situation containing the issue(s), and the future situation containing the solution(s), respectively.

However, even if the three types of SEAM models are different, they can be used as sub-views of one common view. Each of the three model views covers one or few phases of a larger story.

1. *Context*: SEAM modelers analyze the people in their environment exploring their motivations (Goal-Belief)
2. *Conflict*: They show how these motivations map to a desired service design of organizations (Behavior)
3. *Climax*: They show how value is provided by the new organization to customers (Supplier-Adopter-Relationship)
4. *Closure*: They show how the motivations of people are met by the solution to the problem (Goal-Belief)

The expert modeler considers that story telling in SEAM is performed through the hierarchy of service organizations (PureSEAM). He shows how the IT organization helps the business organization, through IT services, and how the business organization serves clients, through business services. However, this interpretation of the story is not accessible for readers unfamiliar with SEAM. In our models we choose one level of the hierarchy (the business organization) and modeled the story of car-maintenance service using seven model instances by showing actors' state changes and explaining the evolution from the "problem" to the "solution" in more detail.

Similar to SEAM models, the two types of i* models (Strategic Dependency and Strategic Rationale) do not allow for the construction of an underlying story because they each represent only one instance of a story. A series of model instances is needed to make the story more explicit. We

combined both types of i^* models into one and made the story more explicit by modeling seven model instances in our proposed model.

One difference between story-telling with SEAM and i^* is that in SEAM elements were updated and even moved between contexts (e.g., car, money, change of actors' beliefs), whereas for i^* they are shown successively by keeping the previous ones in place. With every model instance, SEAM models get more complicated due to an increase in the possible relations between model elements. With every model instance, i^* models get more complex due to an increase in the number of model elements.

In Chapter 6.3, we have presented our third contribution: modeling the story. We have shown how stories are structured in order to explain our storyboard. We have justified why readers focus on understanding stories, based on the “being”, “behaving” and “believing” modeling dimensions. We have presented the design process for several story-telling methods. Finally, we have explained how the SEAM and i^* narrative storyboards were created together with our story-extension of the three types of SEAM models and two types of i^* models into one model structured with seven model instances each. We have created one to three model instances for each story phase by exploring actor's states.

6.4 Contributions to Modelers

The research question of this thesis addressed to modelers is: “How should the modeler create a model so that readers understand the story that the modeler wants to show?”. We answer this question using our work on SEAM and i^* . First, we synthesize the main contributions to modelers who use any modeling notation (e.g., SEAM, i^* , UML, BPMN, ArchiMate). We do so to help them create improved models. Then, we present to modelers a novel modeling process. For SEAM modelers, we describe specific contributions. We conclude with a few trade-offs that modelers should consider.

6.4.1 Recommendations

In Chapters 6.1, 6.2 and 6.3 our recommendations to modelers fall under three categories:

- ***Relation with reality***
 - Use the zone of proximal development between the modeler and the readers' conceptualizations to show concreteness using photos, icons and terminology that characterize actors
- ***Rationale***
 - Show the main question, the options, the criteria and the assessments of the criteria by each actor
 - Use “satisficing” to model options that do not fully satisfy criteria
 - Use “accommodation” to model conflicting interests and consensus
- ***Story***
 - Create model instances for each story phase, e.g., context, conflict, climax and closure
 - For each model instance, explore the actors' states to show change

The first recommendation concentrates on modeling the relation with reality concretely, through the use of the zone of proximal development (the overlap) that is formed between the modeler and the readers' conceptualizations. This concreteness can be attained by modelers by using photos, icons and terminology that characterize actors that are represented in their models. They need to evaluate the trade-off between simplicity of using abstract shapes and complexity of searching for the right photo / icon. The picture will often reveal more meaning than the use of terminology.

The second recommendation focuses on modeling the rationale. This can be achieved by showing explicitly the main question(s), the option(s), the criteria and the assessments of criteria (QOC framework) by decision actors in the model. When the framework is not able to accommodate particular situations in which decisions deal with wicked problems that cannot be tackled rationally, modelers should consider modeling "satisficing" options in order to model choices (or options) that do not fully satisfy all criteria. Similarly, they should consider modeling "accommodations" (or consensus) when dealing with conflicting interests of actors.

The third recommendation centers on modeling the story by using story phases. Modelers might use the story-phases that are particular to their story, not necessarily the context, conflict, climax and closure. They should create a few model instances for each story phase. The number of model instances can be decided by the modeler based on the granularity of changes in actors' states.

6.4.2 Modeling Process

Based on our work on creating improved SEAM and i* models, we provide modelers with a modeling process that helps them improve their models. This process can be followed when creating models with participants, for instance, during workshops. Based on our work on improving the SEAM and i* models, we propose the following steps to modelers to improve their models:

1. Write a short textual description of the story
2. Create an initial model
3. Evaluate the model with a few readers
4. Implement the readers' suggestions
5. Present the model during a workshop
6. Evaluate how participants understand the model at the end of the workshop

Steps 1-4 occur before the workshop, Step 5 occurs during the workshop and Step 6 occurs after the workshop.

The story requires the modeler's interpretation of a situation. For SEAM, we, as modelers, constructed our own story based on personal experience. For i*, we took the textual description of the story published in (Yu E. S., 1997). For step 1, modelers should formulate a clear textual description of the story. Step 2 refers to creating an initial model based on the textual description. Step 3 centers on the evaluation of the model with readers. For step 4, the modeler needs to implement the readers' suggestions that maintain the identity of the modeling notation. In step 5, the modeler presents the model during a workshop with other participants that did not evaluate the model. Finally, in step 6, he gathers feedback from workshop participants to evaluate how they understood the story.

(Weinberg G. M., 1975) gives the following advice:

“To be successful, we must approach complex systems with a certain naïve simplicity. We must be as children, for we have much evidence that children learn most of their more complex ideas in just this manner, first forming a general impression of the whole and only then passing down to more particular discriminations.”

Once the problem is stated in words, instead of immediately creating complex models, the modeler should help readers to form a generic interpretation. Then, he should steer it in the desired direction by using visual cues that enable readers to use analogies, categorizations and generalizations based on their own experiences, but without jumping to conclusions too fast. Focused on the problem-solving aspects, the modeler might neglect to consider whether readers would morally / ethically approve the solution. To be true to himself, the modeler should consider moral questions before arriving at the solution (finalized model), or even while defining the concepts (Gause & Weinberg, 1990).

When evaluating the model with readers (step 3) or with workshop participants (step 6), the modeler will notice that readers frequently arrive at two types of misunderstandings: different names for the same things and same names for different things. When analyzing how people perceive these misunderstandings, the modeler needs to evaluate his models with various audiences (“a foreigner, someone blind or a child” or make himself “foreign, blind or childlike” (Gause & Weinberg, 1990)) in order to identify the possible difficulties for the readers.

The proposed modeling process can be applied for creating models, regardless of the modeling notation (academic or industry) and the modeling scenario.

6.4.3 Observations for SEAM Modelers

We present a few observations that are specific for SEAM modelers.

Although we could have improved separately each type of SEAM model (Goal-Belief, Behavior and Supplier-Adopter-Relationship), we found that we needed to combine and improve all in order to make the story of the considered case (car-maintenance service) as explicit as possible. Other SEAM modelers might consider improving each type of SEAM model separately by modeling other cases.

The SEAM Goal-Belief model is useful for showing the goals and beliefs of actors. With it, it is not possible to model the choice between alternatives, whereas we needed to do so. However, it is possible to represent goals and beliefs for a single alternative. A separate Goal-Belief model is needed for each alternative. From readers, we learned that the modeler needs to explicitly model not only the alternative that was chosen but also the one that was not chosen. This is included in the proposed models but not in a typical Goal-Belief model. Therefore, we extended Goal-Belief modeling with the choice rationale. We recommend that modelers include the rationale in the model based on the Questions, Options and Criteria framework (Chapter 2.3.1) so that readers understand what is the main question, what are the options to answer the question, with what criteria are the options evaluated, and how each actor evaluates each option.

The SEAM Behavior model is useful for showing multiple hierarchical levels. From readers, we learned that showing multiple hierarchical levels produces confusion because of the inclusion of systems as wholes inside systems as composites. Therefore, we modeled only one hierarchical level,

the market with two customers and two suppliers. We did not show in our proposed models the implementation of the (business) services offered by the two dealers at detailed IT levels (containing IT services and processes). Other SEAM modelers might consider modeling other hierarchical levels, such as the business and/or the IT. This extension can be achieved by following the modeling recommendations stated previously, applied at every hierarchical level. We recommend SEAM modelers to model only one level at a time. For more SEAM-inclined readers, more hierarchical levels can be added; but for readers unfamiliar with SEAM, choosing one context at a time is recommended. The terminology specific to SEAM is grounded in concepts such as service, process, whole and composite that readers have difficulties interpreting. These concepts should not be included in the SEAM models. Rather than using “w” and “c” for wholes and composites, modelers should explain only one hierarchical level at a time, in which the distinction between the two is not needed. Wholes and composites can be modeled visually in their respective contexts. SEAM modelers should specify the behavior of a system as a whole with a service, but not name this behavior a service. Similarly, they should specify the behavior of a system as a composite with a process, but not name this behavior a process.

The SEAM Supplier-Adopter-Relationship model is useful for showing, via components, features and values, the relation between a supplier (with its partners, if needed) and a customer. This model uses the two views of SEAM, whole and composite, of the supplier to show the components and features, respectively. Readers found confusing the modeling of the two views of the same entity in the same model, because they did not know if it was the same organization or not. In our model, we show the Delaisse dealer as a whole in order to display the service characteristics, and as a composite when the car is in service, but do not specify the distinction between wholes and composites. There is no visual difference between the two views, except for what is shown inside the boundary. First, the readers understand what happens with the car before it is serviced (the offer / service understanding phase) and only later, do they understand what happens with the car when it is serviced (the service implementation / servicing process phase).

In SEAM, the second story-phase (conflict) corresponds to problem(s), whereas the last story-phase (closure) corresponds to the solution(s). We advocate the need for context (first story phase) about the problem (where does the problem come from). The context can be shown using a SEAM Goal-Belief representation in the same model. We also advocate the need for more detail about the change between the problem and the solution (third story phase); the change can be shown using a SEAM-Supplier-Adopter-Relationship representation in the same model. We recommend that SEAM modelers structure the story based on story phases and that they dissect the composition of models corresponding to each story phase.

6.4.4 Concluding Remarks

Models can be done as sketches, with pen and paper, during workshops, but they are too vague and cryptic for readers when they are read afterwards. Or they can be done using modeling tools after the workshop, but they become too complicated for readers to interpret alone. Our recommendations apply to creating models after the workshop. Modelers need to know the modeling process of the IT notation. They need to be aware of a few trade-offs. The first is between the model’s complexity (consisting of many different elements) and its clarity (readers’ understanding of the story), or between space and understanding. Modelers need to make choices of what to show in their mod-

els. The second is between implicitness (hiding possible outcomes) and explicitness (showing possible outcomes). Modelers need to delimit the problem. The third is between customized (with notation variations for each case / example) and standardized (same notation for all cases) modeling. Modelers need to consider promoting the notation and extending it to other contexts. The usage of a customized notation for every case does not facilitate readers' recognition of it. Modelers could consider giving their models to a professional designer for improvement and then use this improved design to all future modeling cases. The modeler needs to explain to the professional designer the principles that he needs limits of his new design based on the principles of the notation.

6.5 Contributions to Designers

The research question of this thesis addressed to designers is: "How should the designer of modeling notations design a notation so that readers understand the story that the modeler wants to show?". We answer this question using our work on SEAM and i*. First, we synthesize the main contributions to designers who design any modeling notation. We do so to help them design improved modeling notations. Then, we present to designers a novel design process. For SEAM designers, we describe specific contributions. We conclude with a few trade-offs that designers should consider.

6.5.1 Recommendations

In Chapters 6.1, 6.2 and 6.3 our recommendations to designers fall under three categories:

- ***Relation with reality***
 - Identify implicit elements in the modeling notation by evaluating readers' understanding of what elements represent
- ***Rationale***
 - Use visual cues to guide readers' understanding of the solution to the problem
- ***Story***
 - Create models in which different stories are told to elicit readers' understanding of what happens in the model

The first recommendation is a sanity check for improving a modeling notation. In order to design a notation that emphasizes the relation to reality, designers need to identify the elements that are implicit, or difficult to understand, by readers. They can do this by evaluating the readers' understanding of what elements represent. This task is difficult to achieve by the expert modeler, who only uses the notation, as he knows in detail all its intricacies. The designer, who can change the notation, needs to "observe" and "listen" to others interpreting his models. This feedback helps him identify various aspects of the notation that are unclear or confusing to readers. We recommend that designers of modeling notations take into account notation elements that might trigger interpretations different than what modelers want and that they replace those in the notation.

The second recommendation refers to the design of a notation that includes the rationale. Such notation needs to include visual cues that guide readers' understanding of the solution to the problem shown in models created with this notation. These visual cues or graphical symbols can be evaluated with readers in studies that are similar to those performed to improve UML, BPMN, i*, e.g., (Moody, Heymans, & Matulevicius, 2010).

The third recommendation refers to the design of a notation that enables modelers to convey stories. Designers of such notations need to facilitate the creation of models in which different stories are told. It is useful to observe the readers' understanding of what happens in the model, or what the modeler tries to show.

6.5.2 Design Process

Using our work on designing improved SEAM and i^* notations, we provide designers with a design process that helps them improve their notations. This process can be followed when designing new as well as improved notations. It involves readers in the evaluation of models created with these notations. Based on our work on improving the SEAM and i^* notations, we propose the following steps to designers to improve their notations:

1. Elicit required possible stories supported by the modeling notation
2. Match the graphical elements of the modeling notation with the story
3. Evaluate created models with readers to improve the notation

The first step consists in the elicitation of required possible stories, supported by the modeling notation. Designers of notations need to be aware of the possible stories that modelers are able to convey using their notation. Some modeling notations enable only one or a few stories. Other notations might be richer, more loosely defined, thus allowing for more stories or variations of these stories. For instance, the SEAM notation enables modelers to combine the three types of SEAM models (Goal-Belief, Behavior and Supplier-Adopter-Relationship) to show more elaborate stories (e.g., one that reflects the goals, beliefs and actions of actors). Similarly, the i^* notation enables modelers to combine the two types of i^* models (Strategic Dependency and Strategic Rationale) to show more elaborate stories (e.g., one that reflects both the dependencies and the rationale).

The second step centers on matching the graphical elements of the notation with the story. Designers might exploit certain graphical commonalities that readers could interpret based on their common sense, experiences (embodied cognition), behavior, culture or consciousness, as pointed out by (Weinberg G. M., 1975) (e.g., red color for danger, problem or critical element; green color for a functional element; pink / light blue for female / male).

The third step concentrates on the evaluation of the created models with readers to improve the notation. Notation designers should enable modelers to create models with their proposed notation and evaluate these models with readers. By learning about how readers interpret these models, designers can improve their notations.

The above design process can be applied for designing (new or improved) notations, regardless of the application domain (e.g., IT / business).

6.5.3 Observations for SEAM Designers

In this work, we proposed improvements to the SEAM notation rather than create a new notation similar to SEAM. It is important for a SEAM designer to recognize his notation in the models that SEAM modelers create. So, what is the identity of SEAM? What are the notation elements of SEAM that makes it to be recognizable as SEAM and not another notation?

In Figure 76 and Figure 77, we show two examples of models that look similarly visually, but are different: the first model is created in accordance with the identity of SEAM, whereas the second one is not.

In Figure 76, we show the context represented by the market segment shown as a system as a composite. In this context we show two actors: a supplier and a customer, both modeled as systems as wholes. The supplier (placed to the left) offers a service to the customer (placed to the right) via a process. The supplier service, the customer service and the process are all modeled as wholes. The two services participate in a process.

With this example we demonstrated three fundamental SEAM principles: the context needs to be shown, the suppliers need to be positioned to the left and the customers need to be positioned to the right, and services (from systems as wholes) need to be linked to processes (from the context as composite).

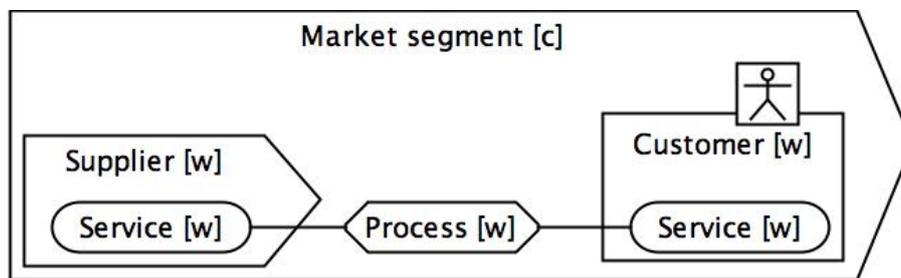


Figure 76: Model that is in accordance with the identity of the SEAM notation

In Figure 77, we show a model that “looks” as if it was created with the SEAM notation, but which is not in accordance with the three SEAM principles. First, the context (larger box including all other model elements) is missing. Second, the customer is positioned to the left and the supplier to the right. Third, the supplier’s service is not linked to the process, but rather the supplier is linked to the process.

Therefore, the three SEAM principles are incorrectly applied. Some model readers and modelers will argue that the model from Figure 77 is a SEAM model because the forms are SEAM like, but the rules of composition of SEAM are violated. This poses the question of what the identity of SEAM is, given that for many observers this model will look like a SEAM model. Relatively few people (e.g., Professor Alain Wegmann, the founder and notation designer of SEAM) will notice that this model is not in accordance with the identity of the SEAM notation.

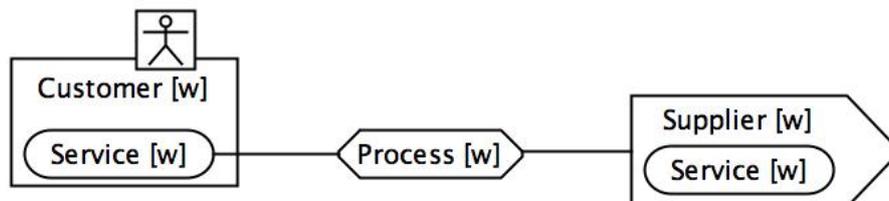


Figure 77: Model that is not in accordance with the identity of the SEAM notation

Other principles of the SEAM notation are:

- services need to be included in systems as wholes

- processes need to be included in systems as composites
- a system can be shown as a whole or as a composite but in the same context (the composite cannot be outside the composite)
- services and processes can be modeled as wholes or as composites

A notation which is not in accordance with these notation principles has lost its identity for some of its observers (e.g., SEAM designers). A detailed description of all SEAM notation principles was provided in Chapter 2.1.1. The challenge is that these principles are difficult to grasp visually by non-SEAM readers. The improvements proposed in this thesis maintain the identity of SEAM, i.e. are in accordance with the above principles. One might consider different levels of a notation's identity, i.e. discuss what to change / keep in a model: for instance, the fact that if most principles (non-negotiable by the modeler) are satisfied, then the notation would still be SEAM. In this case, the expert SEAM modeler (who knows all underlying SEAM principles) would need to compromise the choices he makes in his models for non-SEAM audiences (that do not know any of the underlying SEAM principles). SEAM modelers who are flexible with their notation will still perceive the notation to be SEAM, whereas others who are less flexible will perceive the notation to not be SEAM.

In terms of the practical problem for designers, Professor Alain Wegmann, the creator of the SEAM notation, has thought of SEAM as a tool used for story-telling for several years. However, he has never formalized it. Our contribution for Prof. Wegmann, as a SEAM modeler, is the formalization of story-telling with SEAM, by using the creation of model instances for each of the four story-phases.

6.5.4 Concluding Remarks

When improving modeling notations, designers should maintain their identity. But what constitutes the modeling notation to be that modeling notation? What makes SEAM to be SEAM? What makes i^* to be i^* ? What makes a modeling notation to be that modeling notation (and not another one)? Each notation possesses a graphical vocabulary (set of symbols) together with syntax (set of rules for connecting the symbols) that together form the identity of the notation. This is well known to modeling experts, but not to readers. Readers do not have the knowledge nor they perceive all subtleties of notations. We name implicit these unperceived elements that are difficult to comprehend by readers. We make these elements more explicit, so that readers can interpret the story easier. We make the story that the modeler wants to show understandable to readers using graphical cues. The readers' interpretation of the story might change based on their knowledge, experience and imagination. Regardless of this change, the modeler needs to be able to show a story based on the model's structure and content so that readers understand this story. For an expert, each aspect of the model needs to be coherent with the entire story and is in accordance with the principles of the underlying modeling notation. Some of those aspects might be considered as minor, mere details by readers that are unfamiliar with the notation. As we have seen with modeling with SEAM and i^* , creating a SEAM model of the i^* case is relatively easy for SEAM modelers. It demands that the modeler applies SEAM to another situation. Improving i^* to account for the lack of elements (e.g., the main question) needs a deeper reflection from the modeler. This is because of the numerous constraints of the modeling notation, constraints that are specific and different from one notation to another.

Necessarily, this work is related to hermeneutics, or the study of interpretation (Winograd & Flores, 1986). As expected for qualitative interviews, different people interpret models in different ways. Some people might also interpret the same model differently at different moments in time. Either assuming that the understanding of the model lies either within the model itself or within the neural capacity of the reader (Maturana & Varela, 1987) is not sufficient. The modeler needs to establish the relation between the model and the reader. The understanding of the model by the reader is an emergent property of the model. As (Weinberg G. M., 1975) mentions, the emergent properties emerge as a relation between the observer and the observed.

6.6 Discussions of Recommendations with SEAM Modelers

We discussed the recommendations proposed in this thesis with six SEAM modelers who evaluated the relevance of our recommendations for them:

- Prof. Alain Wegmann – Professor at EPFL, SEAM designer and Consultant,
- Dr. Gil Regev – Senior Researcher at EPFL and Knowledge Manager at ITECOR (a consulting company in Switzerland),
- Mr. Didier Rey – Vice-Presidency of Information Systems Delegate at EPFL,
- Mr. Giorgio Anastopoulos – Head of Information Systems Architecture at EPFL,
- Mr. Olivier Hayard – Vice-President Head of Knowledge Management at ITECOR, and
- Mr. Gaël de Fourmestraux – Head of Geneva Office at ITECOR.

These modelers use SEAM during workshops that they organize with business and IT stakeholders. We presented the models created in this thesis and asked if they are useful for them to create other models with other people.

SEAM modelers appreciated that the research method used in this thesis is particularly useful for business strategy communication. They found useful to use different notations to prepare a workshop, to create models during the workshop, and to communicate the results to other stakeholders in the organization. One modeler found not useful to impose a notation when working with people unfamiliar it during a workshop. He suggested SEAM modelers should not use SEAM when working with readers unfamiliar with SEAM. He argued that, in this case, it is better to create an ad-hoc notation that the audience understands. It is important for the SEAM notation designer to maintain the identity of his notation so that modelers and readers recognize it. This core identity of the notation is beyond its visual notation. The constraint placed on this thesis by the SEAM designer was to maintain the identity of SEAM - not invent another notation. This is because the designer wants people to think about their problems through the use of his notation. Designing an improved SEAM demands for less flexibility to try possible changes, than designing a new notation. Therefore, this research is mostly useful for designers who would like to maintain the identity of their notation. Designers investigating new notations might find the approach from this research less useful.

SEAM modelers acknowledged the importance of identifying implicit elements of the SEAM notation by working closely with readers unfamiliar with the notation (especially a large number of them). One of the most challenging aspects of their daily practice is to convince model stakeholders of what elements represent. They welcomed our investigation of people's perceptions of model elements (photos, icons and terminology). It was useful for them to learn about the elicitation of im-

pllicit elements by interviewing readers and hear about our SEAM models' weaknesses captured based on readers' reactions to elements that are difficult to understand.

The six SEAM experts, found useful the possibility of telling different stories using the same model. Therefore, for them, one of the useful aspects of this work is the search for the abstraction principles underlying a notation, principles that would allow the instantiation to any situation. In our research, we learned that modelers need to choose where to place their model on the axis ranging from concreteness, preferred by readers, to abstraction, preferred by experts. They recognized the merit of identifying the readers' need to perceive a coherent story in models they are presented with. This can be achieved by putting together different views rather than showing them separately.

To conclude, it is important for the modeler to create correct SEAM models after the workshop, as well as for participants to understand simple SEAM models during the workshop. Non-SEAM readers (participants), who take part in the early requirements of a project, should be presented with easy-to-understand models, in which the identity of the modeling notation is loosely defined, i.e. the models should facilitate story-telling but should not be fully formalized. These models should help to open up discussions in which the perceptions of workshop participants about the characteristics of the project are elicited. They are useful to enable workshop participants to agree on their perceptions about what issues are to be addressed. Cleaner models can be used to communicate to other stakeholders in the organization, at the end of the workshop. These models are both graphically and conceptually positioned in between the loosely defined ones, done with pen and paper during the workshop, and the formal ones created with professional tools by the modeler. The formalism preferred by expert SEAM modelers should not impede the collaboration between readers.

6.7 Limitations

In this research we help modelers create models that are understandable by readers, rather than teach modeling notations to readers. Teaching requires continuous work with readers and needs to be performed in class or during a series of workshops. Our approach is centered on the relations between the modeler, the model, the readers and the story, rather than solely improving the model. We consider that modelers convey stories in their models; hence, our model evaluation-metric is the difference between the story the modeler wants to convey and the one readers interpret. We consider that readers and modelers interpret their observed reality. We exploit the relation between the modeler and the reader with the zone of proximal development between their conceptualizations.

Our contributions are based on interviewing one hundred and twenty model readers (the vast majority European), unfamiliar with the modeling notation. If we had interviewed other readers, familiar with the modeling notation, they might have found it easy to understand the story. When working with readers familiar with the modeling notation, the modeler could draw sketches to discuss an idea, whereas when working with readers unfamiliar with the notation he would need to create cleaner models by using professional tools. Our readers only interpreted the models, but did not participate actively in creating them.

We did not consider culture as a separate factor to analyze readers' interpretation of the modeler's story. Readers from different cultures might interpret models differently, e.g., readers of Arabic background might prefer reading the model form right to left, readers of Asian background might

focus on roles, superiority and hierarchy rather than open collaboration. In addition, people who do not commonly work with graphical models might need more time to interpret them.

In our interviews, our readers first matched the model elements with their experience. To predict what the model entailed, they made sense of the models with their existing knowledge. Readers who experienced a similar situation as the one presented in the models were able to relate well with the actors in the context shown (self-referentiability). However, some readers interpreted the model differently than the modeler wanted, as they supposed how the situation should have happened. Readers unfamiliar with the situation were generally receptive to the model, whereas those who were familiar with it related their experience against the model and were less receptive to learn.

We modeled only one hierarchical level of SEAM: the market segment with two customers and two suppliers (one system seen as a composite with four systems seen as wholes). We did not model the dealer(s) as composite(s) and we did not show the service implementation in detail. The car was shown at the dealer that was chosen, without the component sub-systems of the dealer that offer services to implement the process. We tested with readers if they understand the view of the dealer as a composite but noticed confusion as they asked about the differences between the two views in terms of what can be shown and what needs to be shown. We learned to show readers only one level at a time, instead of multiple (e.g., one composite system within another composite system).

Modelers need to create models for their target reader-audiences (e.g., readers unfamiliar with the notation). They can benefit from involving these readers in the model creation process. Similarly, designers need to design modeling notations for their target modeler-audiences. They can benefit from involving these modelers in the notation design process. The process of implementing changes in an improved model based on an improved notation is time consuming and demands continuous experimentation.

Modelers use notations to communicate to readers unfamiliar with these notations, for instance discuss business and IT strategy during workshops. If these (IT) modelers would create models with these IT notations, (non-IT) readers would not understand them, because they were originally thought of as applicable to software engineers who work with dedicated tools (e.g., SeamCAD, i* software). There are no automatic tools designed for IT modelers to create models for non-IT readers. The creation of an improved notation requires an iterative process of model creation, in which modelers reflect about and evaluate their choices of graphical elements (e.g., what photos, icons and terminology to use and how they relate) with readers. Most tools (e.g., SeamCAD, TradeYourMind) restrict the definition of certain graphical elements based on the context. They enable the types of relationships between elements, based on the fundamental principles of the notation.

6.8 Summary

In Chapter 6, we have presented the contributions of our research in relation to the theoretical background. We have discussed our first contribution, modeling the relation with reality, by arguing on the need of using the zone of proximal development between the modeler and the readers' conceptualizations to show concreteness using appropriate photos, icons and terminology. The model elements should be based on readers' conceptualization rather than the modeler's, but the way they are related should be based on the identity of the modeling notation. We have presented our second contribution, modeling the rationale, showing the main question, the options, the criteria and the

assessment of criteria in the model by using “satisficing” to model options that do not fully satisfy criteria and “accommodation” to model conflicting interests and consensus. We extended the Questions, Options and Criteria framework with five-scale assessments (instead of binary) for two actors (instead of one). We have elaborated on our third contribution, modeling the story, by designing storyboards with seven model instances based on four story-phases: context, conflict, climax and closure, in which we explored the actors’ states to show change. Then, we have presented our contributions to modelers, including recommendations and a modeling process, followed by our contributions to designers of modeling notations, including recommendations and a design process. We have concluded with a discussion of the limitations of our contributions.

Chapter 7 Conclusions and Future Work

7.1 Achieved Results

IT modeling notations are used in all areas of information systems practices, from eliciting early requirements to designing enterprise architectures and IT systems. IT people create models with these notations in order to communicate with business (non-IT) people. The models are created and used by IT people and show the collaboration between business and IT people. The problem is that these models are not understandable by business people.

The main purpose of modelers working with such notations is to create models that enable readers to interpret the story that the modeler wants to convey. Put simply, the model should reduce the misalignment between the stories that the modeler conveys and the readers interpret, respectively.

Our research is inter-disciplinary and includes references to: general systems thinking (Weinberg & Weinberg, 1988), systems inquiry (Banathy & Jenlink, 2004), graphical argumentation (Buckingham Shum, MacLean, Bellotti, & Hammond, 1997), interpretation of reality (Maturana & Varela, 1987), learning (Vygotsky, 1997), story-telling (McCabe & Peterson, 1984), evaluation of modeling notations (Moody D. L., 2010), design science research (Hevner, March, Park, & Ram, 2004) and social science research (Booth, Colomb, & Williams, 1995), (Yin, 1994), (Bhattacharjee, 2012), and (Kvale & Brinkmann, 2008)). It enabled us to recommend that modelers create models by considering (1) the relation with reality, by using the zone of proximal development between the modeler and readers' conceptualizations that enables readers to learn from the modeler's conceptualization, (2) the rationale, by using argumentation (leading to the choice) by showing questions, options, criteria and their assessments, and (3) the story, by designing storyboards with model instances by using story phases.

The originality of our research lies in understanding readers' conceptualizations in order to create models and design notations. We have created models that were evaluated by one hundred twenty readers. The readers' suggestions were instrumental in identifying difficult to comprehend elements and in making them more explicit. The modeler familiar with the modeling notation can seldom identify these implicit elements as he already knows them, but his readers do not. It is useful to evaluate models with people unfamiliar with the modeling notation.

We have proposed recommendations for creating models, based on improving two visual notations, SEAM and i*, using two examples: car-maintenance service and meeting scheduler. First, we modeled the relation with reality by using the zone of proximal development between the modeler and the readers' conceptualizations to show concreteness using photos, icons and terminology that characterize actors. Second, we modeled the rationale and choice in SEAM and i* more explicitly by showing the main question, the options, the criteria and the assessment of criteria by actors. We discussed the satisficing of options that do not fully satisfy criteria and the accommodation of con-

flicting interests to model consensus. Third, we created SEAM and i* storyboards with seven model instances each, based on four story-phases: context, conflict, climax and closure in which we explored actors' states to show change.

By using a story, we merged the three SEAM model types and two i* model types into one, respectively. To facilitate story-telling, we created a sequence of model instances for each model. As a result, we reduced the number of model types but increased the number of model instances. Instead of multiple models (different views) of the same situation, we propose one model that illustrates a story (one coherent view) that readers can relate to (e.g., instead of fourteen diagram types in UML we propose to create one that enables story-telling).

We have also proposed recommendations for designing notations. First, we designed the relation with reality by identifying implicit elements in the modeling notation by evaluating readers' understanding of what elements represent. Second, we designed the rationale by using visual cues to guide readers' understanding of the problem. Third, we designed the story by creating models in which different stories are told to elicit readers' understanding of what happens in the model.

7.2 Future Developments

We study modeling notations from three perspectives: motivation, behavior and value (Table 11). We recommend that modelers, who use other modeling notations (e.g., UML, BPMN, ArchiMate, OPM, Use Case Map, BMM, e3-value), apply our recommendations and modeling process to improve their models.

Modeling notations	Motivation	Behavior	Value
Systemic Enterprise Architecture Method	x	x	x
i*	x	x	
Object Process Methodology and Notation		x	
Design and Engineering Methodology for Organizations		x	
Soft Systems Methodology + Rich Pictures	x	x	x
Unified Modeling Language		x	
Business Motivation Model	x		
Use Case Map		x	
Business Process Modeling Notation		x	
e ³ -value			x
ArchiMate		x	

Table 11: Modeling notations and corresponding topics

As we have seen in this work, each modeling notation supports several model types. To improve separately each model type (e.g., SEAM Goal-Belief, SEAM Behavior, SEAM Supplier-Adopter-Relationship), or a combination of several model types (e.g., similar to the present work of combining the three SEAM models and the two i* models into one), modelers can apply our recommendations and modeling process.

We also suggest that modelers apply them to other contexts as well: for instance, in the fields of organizational strategy, service design, enterprise architecture, and business and IT alignment. Modelers can create models to show how decisions are taken and how strategies are defined in organizations. The explanation of the rationale is essential in strategy communication. Modelers should consider specific organizational contexts and work with workshop participants to develop

improved models. To test the readers' comprehension, modelers can explore the readers' interpretations of textual models compared to those of graphical models.

In our research, we considered only one SEAM hierarchical level, the market segment with two customers and two suppliers. SEAM modelers can show other levels (composites inside composites) and more refinements (of wholes into composites), for instance business (service specification with business services and processes) and/or IT (service implementation with IT services and processes). This extension can be achieved by following our modeling recommendations for every hierarchical level. SEAM modelers may also consider modeling conflicting interests of individual actors and groups.

Modelers can explore the creation of more / less model instances for their stories. They can develop these instances by considering other story phases than context, conflict, climax and closure, to construct stories that are differently structured, e.g., with multiple conflicts or climax, with multiple possible closures. They can explore scenario-building for story-telling inspired by movies.

For designers, we recommend they also use our recommendations and design process to design and improve their notations (Table 11). Our initial presentation of several other modeling notations (Appendix 6) can be used as a starting point. Designers could consider the design of new notations connecting different fields of research, e.g. business and IT, requirements engineering and implementation, or organizational design and psychology. Designers can also continue the refinement of principles that capture the identity of the notation, that we started with our research process.

For the continuation of our work on improving the SEAM notation, we recommend the collaboration between a SEAM modeler and a professional graphical designer (illustrator) to explore other design possibilities that would make the SEAM notation more understandable for readers. The designer would need to consider an in-depth understanding of SEAM principles (e.g., modeling the system boundary) and various instantiations to real life examples. He might explore new modeling principles as well.

Researchers can also improve notation-evaluation frameworks using our recommendations. Specifically, they can evaluate the degree to which a notation has the capacity to show explicitly the relation with reality, the rationale and the story. For instance, researchers can consider, as a notation evaluation metric, the readers' understanding of the entire model (the modeler's story represented in model instances). This we find more significant for the communication between the modeler and the readers, rather than the analysis of the cognitive effectiveness of individual graphical symbols within the model (for each model instance).

Glossary

Conceptualization – explanation of reality observed by someone (e.g., modeler, reader).

Designer – person who designs a modeling notation. He analyzes how modelers create their models using a notation. His goal is to help modelers develop models that readers can understand.

IT designer – designer who designs IT notations.

Entity – element or concept from the universe of discourse or observed real-life.

Graphical symbol / visual cue – notation element used by the modeler to show a story in a model.

Implicit / Explicit – difficult /easy to understand by readers; unnoticed / noticed by readers in the model.

Improvement – change of the model to make it more understandable for readers, i.e. so that readers interpret the story that the modeler wants to show.

Interview – meeting with people, unfamiliar with the notation, to elicit implicit elements in models.

Meta-model – defines concepts, their relations and semantics; a model is an instance of a meta-model if it respects the structure defined by the meta-model.

Method – set of practices, tools, procedures, techniques, rules, or processes used by modelers to engage in an inquiry.

Methodology – study of how research is done; principles that guide research practices; set of working methods and their study.

Model – graphical representation, based on a notation, of a conceptualization of reality of a modeler that is used to show a story to an audience of readers. It is a schematic description of something, especially a system or phenomenon that accounts for its properties and is used to study its characteristics and show its construction (or appearance) and behavior (action). A model is used to show a story of a modeler. A model instance reflects a section of the story. A model element is a graphical symbol shown in a model instance.

Modeling –creation of models useful to understand the observed reality by showing more explicitly what is implicit.

Modeler – person who creates a model, the author of the model.

IT modeler – modeler who uses an IT notation to create models.

Notation – set of graphical symbols (visual vocabulary) and composition rules (visual grammar) used to represent a model based on a method.

IT notation – notation used in IT, such as SEAM, i*, UML, BPMN, ArchiMate.

Rationale – analysis of underlying reason(s) or belief(s) contributing to a decision taken by a person.

Reader – person who interprets a model through his own perceptions, usually unfamiliar with the modeling method, the observer.

Research investigator – person who carries out interviews to evaluate models developed with modeling notations; interviewer. His goal is to gather feedback from interviewees (readers) to evaluate if they understand the story that the modeler wants to show. The research investigator is the thesis author.

System – set of interconnected components working together as a whole, a human perception of reality, a viewpoint or “a way of looking at the world” (Weinberg G. M., 1975).

Service – the value that a client expects with respect to what the supplier offers. In SEAM is defined as “an external observable behavior of a system as a whole” (Wegmann A. , 2016). It was considered in the scientific literature as the application of specialized resources (including competences, skills, knowledge) for the benefit or to create value for another (system) or the system itself (Vargo, S. L. and Lusch, R. F., 2004). It consists of service offering and service implementation. It can be defined as well as “a set of related software functionalities that can be reused for different purposes, together with the policies that should control its usage” (Wikipedia, 2015) or “an application of resources for the benefit of a customer” (Maglio et al., 2009) or “a means of delivering value to customers by facilitation outcomes that customers want to achieve without the ownership of specific costs and risks” (ITIL, 2014) or “a specific task that the work system helps the consumer accomplish” (Saxena, A. B. and Wegmann, A., 2014).

Service offering – service that is provided to the customer, without the implementation details.

Service implementation – details behind how the service is provided to the customer.

System – configuration of people, technologies, and other resources that interact with other systems to create common value to an external customer. Examples include families, cities, companies, etc. A system can be represented as a whole showing the service offering, or as a composite showing the service implementation between other sub-systems.

Stakeholder – person with an interest or concern in the model, e.g. workshop participants.

Story – sequence of events shown in the model that is based on the modeler’s conceptualization of the observed reality and is interpreted by readers; what they observe in the model.

Story-telling – act of conveying a story in a model by revealing elements that encourage the reader’s imagination. The modeler seeks to show a story through his model.

Story-boarding – modelling of a flow of user’s activities so that they can be reviewed and evaluated by designers and users.

Bibliography

- Amyot, D. (2003). Introduction to the User Requirements Notation: Learning by Example. *Computer Networks*, 42(3), 285-301.
- Amyot, D., & Gunther, M. (2011, 5). User Requirements Notation: the First Ten Years, the Next Ten Years. *Journal of Software*, 6(5), 747-768.
- ArchiMate*. (2009). Retrieved 4 4, 2016, from The Open Group: http://pubs.opengroup.org/architecture/archimate-doc/ts_archimate/
- Arning, K., & Ziefle, M. (2009). It's a Bunch of Shapes Connected by Lines: Evaluating the Graphical Notation System of Business Process Modelling Languages. *International Conference on Work With Computer Systems*. Beijing.
- Ashby, W. R. (1961). *Introduction to Cybernetics*. New York, NY, U.S.A.: Wiley.
- Audi, R. (1999). *The Cambridge Dictionary of Philosophy*. Cambridge University Press.
- Ausubel, D. P., Hanesian, H., & Novak, J. D. (1978). *Educational Psychology: A Cognitive View*. Holt, Rinehart and Winston.
- Baddeley, A. (1994). The Magical Number Seven: Still Magic after All These Years? *Psychological Review*, 101(2), 353-356.
- Bajić-Bizumić, B., & Wegmann, A. (2015). *Animation-Based Service Specification, Verification and Validation*. Lausanne, Vaud, Switzerland: EPFL.
- Bajic, B., Petitpierre, C., Wegmann, A., & Tri, D. Q. (2012). From Business Services to IT Services by Capturing Design Decisions. *International Symposium on Business Modeling and Software Design*. Geneva.
- Baker, M. J. (2000). Writing a Literature Review. *The Marketing Review*, 1(2), 219-247.
- Balabko, P., & Wegmann, A. (2002). *From the RM-ODP to the Formal Behavior Representation*. (P. F. Specification, Ed.) Kluwer Academic Press.
- Balabko, P., & Wegmann, A. (2005). *Situation-based Modeling Framework for Enterprise Architecture*. Lausanne: EPFL.
- Banathy, B. (2014). *A Taste of Systemics*. Retrieved from <http://www.iss.org/taste.html>
- Banathy, B. H., & Jenlink, P. M. (2004). *Systems Inquiry and its Application in Education - Handbook of Research on Educational Communications and Technology*. Bloomington: Lawrence-Erlbaum.
- Bar, M., & Neta, M. (2006). Humans Prefer Curved Visual Objects. *Psychological Science*, 17(8), 645-648.
- Barsalou, L. W. (2008). Grounded Cognition. *Annual Review Psychology*, 59, 617-645.
- Bertin, J. (1973). *Sémiologie Graphique: Les Diagrammes - Les Réseaux - Les Cartes*. Gauthier-Villars Mouton & Cie.

- Bertrand, G., & Schmitt, M. (2006). Business Service Network Design: From Business Model to an Integrated Multi-Partner Business Transaction. *E-Commerce Technology*. IEEE.
- Beyer, H., & Holtzblatt, K. (1995). *Apprenticing - with the Customer*.
- Beyer, H., & Holtzblatt, K. (1997). *Contextual Design: Defining Customer-Centered Systems*. Morgan Kaufmann.
- Bhattacharjee, A. (2012). *Social Science Research: Principles, Methods, and Practices*.
- Biederman, I. (1987). Recognition-by-Components: A Theory of Human Image Understanding. *Psychological Review*, 94(2), 115–147.
- Bochmann, G. (2016, 07 01). *Model-Based Design and Verification of Distributed Real-Time Systems*. Retrieved from University of Ottawa: <http://www.site.uottawa.ca/~bochmann/ELG7187C/CourseNotes/BehaviorModeling/Petri-nets/index.html>
- Booch, G. (1994). *Object-Oriented Analysis and Design with Applications*. Redwood City, California, U.S.A.: Benjamin Cummings Publishing.
- Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *Unified Modeling Language User Guide*. Addison-Wesley Professional.
- Booth, W. C., Colomb, G. G., & Williams, J. M. (1995). *The Craft of Research*. Chicago and London: The University of Chicago Press.
- Brestovansky, D. (2008). *Exploring Textual Modeling Using the Umple Language*. Ph.D. Thesis, University of Ottawa, Ottawa-Carleton Institute for Computer Science.
- Britton, C., & Jones, S. (1999). The Untrained Eye: How Languages for Software Specification Support Understanding by Untrained Users. *Human Computer Interaction* 14, 14, 191–244.
- Brooks, K., & Quesenbery, W. (2010). *Storytelling for User Experience*. Rosenfeld Media.
- Brown, T. (2008, 6). *Design Thinking*. Retrieved 4 4, 2016, from Harvard Business Review: <https://hbr.org/2008/06/design-thinking>
- Buckingham Shum, S. J., MacLean, A., Bellotti, V. M., & Hammond, N. V. (1997, 2). Graphical Argumentation and Design Cognition. *Human Computer Interaction*.
- Buhr, R., & Casselman, R. (1996). *Use Case Maps For Object-Oriented Systems*. Englewood Cliffs: Prentice Hall.
- Caire, P., Genon, N., Heymans, P., & Moody, D. L. (2013). Visual Notation Design 2.0: Towards User Comprehensible Requirements Engineering Notations. *International Conference on Requirements Engineering*.
- Callahan, S. (2016). *Putting Stories to Work - Mastering Business Storytelling*. Melbourne, Australia: Pepperberg Press.
- Carrupt, B., & Wegmann, A. (2016). Retrieved from SeamCAD: <http://lams.epfl.ch/seamcad/versions/4/>
- Chan, Y., & Reich, B. (2007). IT Alignment: What Have We Learned? *Information Technology*, 22, 297-315.
- Checkland, P. (1990). *Soft Systems Methodology in Action*. Chichester, UK: Wiley.
- Checkland, P. (1990). *Systems: Concepts, Methodologies and Applications (SCMA)*. Wilson.
- Checkland, P. (1999). *Systems Thinking, Systems Practice*. John Wiley & Sons.
- Checkland, P., & Holwell, S. (1998). *Information, Systems and Information Systems - Making Sense of the Field*. Chichester, UK: Wiley.

- Chen, W., & Hirschheim, R. (2004). A Paradigmatic and Methodological Examination of Information Systems Research from 1991 to 2001. *Information Systems Journal*, 14(3), 197–235.
- Chung, S., Won, D., Baeg, S. H., & Park, S. (2009). Service-oriented Reverse Reengineering: 5WIH Model-driven Re-documentation and Candidate Services Identification. *International Conference on Service-Oriented Computing and Applications* (pp. 1-6). Taipei: IEEE.
- Citrin, W. (1996). Strategic Directions in Visual Languages Research. *ACM Computing Surveys*, 24(4).
- ConceptDraw. (2016, 07 01). *Business Process Modeling Notation Template*. Retrieved from ConceptDraw: <http://www.conceptdraw.com/How-To-Guide/business-process-modeling-notation-template>
- Corporate Memory Systems, I. (1993). *QuestMap*. Retrieved from CMSI: <http://www.cmsi.com/info/>
- Dardenne, A., van Lamsweerde, A., & Fickas, S. (1993). Goal-directed Requirements Acquisition. *Science of Computer Programming*, 20, 3-50.
- Dardenne, A., van Lamsweerde, A., & Fickas, S. (1993). Goal-Directed Requirements Acquisition. *Science of Computer Programming*, 20, pp. 3-50.
- Davies, I., Green, P., Rosemann, M., Indulska, M., & Gallo, S. (2006). How Do Practitioners Use Conceptual Modelling in Practice? *Data and Knowledge Engineering*, 58, 358-380.
- Davodeau, E. (n.d.). *La Technique Du Dessin en BD*. Retrieved from Le Dossier Du Mois: <http://www.millebulles.com/ens/dossiers/dossier13/dossier13.php?IDdos=13>
- D'Souza, D. F., & Wills, A. C. (1999). *Objects, Components and Frameworks With UML. The Catalysis Approach*. AddisonWesley.
- Dori, D. (2002). *Object-Process Methodology. A Holistic Systems Paradigm*. Springer.
- Dupont, L. (2016, 05 24). Les Contes de Fées et Leurs Illustrations. (E. Pré-de-la-Tour, Ed.) Lausanne, Vaud, Switzerland: Carl G. Jung Institut Zürich.
- Edelkoort, L. (n.d.). *The Pop-Up Generation Design between Dimensions*. Amsterdam, 2015: BIS Publishers.
- Endres, A. (1996). A Synopsis of Software Engineering History: The Industrial Perspective. In A. Brennecke, & R. Keil-Slawik, *History of Software Engineering*.
- Etzlinger, L., Castori, P., & Wegmann, A. (2016). Retrieved from TradeYourMind: <http://www.tradeyourmind.com/>
- Evans, A., France, R., Lano, K., & Rumpe, B. (1999). The UML as a Formal Modeling Notation. *The Unified Modeling Language, "UML" 1998: Beyond the Notation*, pp. 336-348.
- Falleri, J.-R. H. (2006). *Towards a Traceability Framework for Model Transformations in Kermeta*.
- Fang, X.-l. (2014, 6). The Research on Electronic Commerce Model Based on 5W2H: Taking Campus E-commerce Value Creation Process as an Example. *Journal of Mianyang Normal University*.
- Fischer, M. (1999). Process Improvement by Poka-Yoke. *Work Study*, 48(7), pp. 264-266.
- Fowler, M. (2004). *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. Addison-Wesley Professional.
- Gause, D. C., & Weinberg, G. M. (1989). *Exploring Requirements - Quality before Design*. New York, NY, USA: Dorset House Publishing.
- Gause, D. C., & Weinberg, G. M. (1990). *Are Your Lights On? How to Figure Out What The Problem Really Is*. New York, U.S.A.: Dorset House Publishing.

- Genon, N., Amyot, D., & Heymans, P. (2010). Analysing the Cognitive Effectiveness of the UCM Visual Notation. *International Conference on System Analysis and Modeling* (pp. 221-240). Berlin: Springer-Verlag.
- Genon, N., Amyot, D., Heymans, P., & Moody, D. (2010). *Applying the "Physics" of Notations to (URN) Use Case Maps*. University of Namur.
- Genon, N., Heymans, P., & Amyot, D. (2010). Analysing the Cognitive Effectiveness of The BPMN 2.0 Visual Notation. *Proceedings of the Third International Conference on Software Language Engineering*. Eindhoven.
- Glaser, B. G., & Strauss, A. L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Adline, U.S.A.
- Goldstein, N., & Alger, J. (1992). *Developing Object-Oriented Software for the Macintosh - Analysis, Design, and Programming*. Reading, MA, U.S.A.: Addison-Wesley.
- Golnam, A., & Wegmann, A. (2013). *Problem Structuring with the Systemic Enterprise Architecture Method: Representation of Systems and Value in Business Contexts and Integration with Operations Research Methods*. Ecole Polytechnique Fédérale de Lausanne. Lausanne: EPFL.
- Golnam, A., Tapandjieva, G., Viana, S., & Wegmann, A. (2011). Problem Structuring Methods in System Dynamics Modeling - A Cognitive Fit Perspective. *International Conference of The System Dynamics Society*. Washington.
- Goodman, N. (1968). *Languages of Art: An Approach to a Theory of Symbols*. Indianapolis, U.S.A.: Bobbs- Merrill Co.
- Gordijn, J. (n.d.). e3-value in a Nutshell. *International Workshop on e-business Modeling*. Lausanne: HEC Business School.
- Gordijn, J., Yu, E., & van der Raadt, B. (2006). e-Service Design Using i* and e3-value Modeling. *Software*, 23(3), pp. 26-33.
- Gourlay, S. (2003). *The SECI Model of Knowledge Creation: Some Empirical Shortcomings*.
- Green, T. R., & Petre, M. (1996). Usability Analysis of Visual Programming Environments: A 'Cognitive Dimensions' Framework. *Visual Languages and Computing*, 7, 131-174.
- Green, T. R., Blandford, A. E., Church, L., Roast, C. R., & Clarke, S. (2006). Cognitive Dimensions: Achievements, New Directions and Open Questions. *Visual Languages And Computing*, 17(328-365).
- Green, T., & Blackwell, A. (2016). *Cognitive Dimensions of Notations*. Retrieved 4 4, 2016, from Resource Site: <http://www.cl.cam.ac.uk/~afb21/CognitiveDimensions/index.html>
- Greenberg, S., Carpendale, S., Marquardt, N., & Buxton, B. (2012). *Sketching User Experiences - The Workbook*. Elsevier.
- Grier Miller, J. (1978). *Living Systems*. Niwot: University Press of Colorado.
- Groenewegen, J., Hoppenbrouwers, S., & Proper, E. (2010). Playing ArchiMate Models. *Enterprise, Business-Process and Information Systems Modeling*, 50, 182-194.
- Gruber, T. (2009). Ontology. *Encyclopedia of Database Systems*.
- Hale, A. (2008, 9 17). *How to Structure a Story: The Eight-Point Arc*. Retrieved 4 4, 2016, from <http://www.dailywritingtips.com/how-to-structure-a-story-the-eight-point-arc/>
- Halpin, T. (2005). *ORM 2 Graphical Notation*. Neumont University.
- Harel, D., & Rumpe, B. (2004). Meaningful Modeling: What's the Semantics of "Semantics"? *IEEE Computer Society*.

- Hauser, J. R., & Clausing, D. (1988). *The House of Quality*. Harvard Business Review.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004, 3). Design Science in Information Systems Research. *Management of Information Systems Quarterly*, 28(1), 75-105.
- Hitchman, S. (2002). The Details of Conceptual Modelling Notations are Important - A Comparison of Relationship Normative Language. *Communications of the AIS*, 9(10).
- Hodgson, A. M. (1992). Hexagons for Systems Thinking. *European Journal of Operational Research*, 59(1), 220-230.
- Horkoff, J., & Yu, E. (2016, 4 4). *i**. Retrieved from Aachen University: <http://istar.rwth-aachen.de/tiki-index.php?page=iStarQuickGuide>
- IDEO. (n.d.). *Design Thinking for Educators Toolkit*. Retrieved from <https://www.ideo.com/work/toolkit-for-educators>
- ISO. (2016). *The International Standard Organization*. Retrieved from <http://www.iso.org/iso/home.html>
- ITIL. (2016). *The Information Technology Infrastructure Library*. Retrieved from <http://www.itil-officialsite.com/>
- Jacobson, I., Christerson, M., Johnsson, P., & Overgaard, G. (1997). *Object-Oriented Software Engineering: A Use Case Driven Approach*. Reading, MA, U.S.A.: Addison-Wesley.
- Jang, S., & Woo, W. (2005). 5W1H: Unified User-Centric Context. *International Conference on Ubiquitous Computing*. IEEE.
- Johnson, P., Ekstedt, M., & Jacobson, I. (2012). Where's the Theory for Software Engineering? *IEEE Computer Society*.
- Jones, A., & Weiwei, A. (2012, 1 4). *The Art of Destruction / The Destruction of Art*. Retrieved 4 4, 2016, from Image Object Text: <http://imageobjecttext.com/2012/01/04/the-art-of-destruction-the-destruction-of-art/>
- Jonkers, H., Lankhorst, M., & Proper, E. (2011). ArchiMate for Integrated Modelling Throughout the Architecture Development and Implementation Cycle. *Commerce and Enterprise Computing (CEC)*, 294 - 301.
- Kühn, T. (2012). *The Structure of Scientific Revolutions*. U.S.: University of Chicago Press.
- Kautzer, K. (2012, 3 19). *5 Stages of Storytelling*. Retrieved 4 4, 2016, from <http://writeshop.com/5-stages-of-storytelling/>
- Kim, J., Hahn, J., & Hahn, H. (2000). How Do We Understand a System With (So) Many Diagrams? Cognitive Integration Processes in Diagrammatic Reasoning. *Information Systems Research*, 11(3), 284–303.
- Klems, B. A. (2016, 1 16). *7 Steps to Creating a Flexible Outline for Any Story*. Retrieved 4 4, 2016, from <http://www.writersdigest.com/online-editor/7-steps-to-creating-a-flexible-outline-for-any-story>
- Kolber, A. B. (2010). *The Business Motivation Model - Business Governance in a Volatile World*. The Business Rules Group.
- Kraan, W. (2016, 07 01). *ArchiMate Introduction*. Retrieved from <http://www.slideshare.net/emergingpractices/archimate-overview>
- Krogstie, J. (2006). Process Models Representing Knowledge for Action: A Revised Quality Framework. *European Journal of Information Systems*, 15, 91–102.
- Krogstie, J. (2012). *Model-Based Development and Evolution of Information Systems - A Quality Approach*. London, U.K.: Springer.
- Krogstie, J. S. (2003). *Information Systems Engineering - Conceptual Modeling in a Quality Perspective*. Kompendiumforlaget, Trondheim.

- Kvale, S. (1996). *Inter Views: An Introduction to Qualitative Research Interviewing*. Thousand Oaks, California, U.S.A.: Sage.
- Kvale, S., & Brinkmann, S. (2008). *Interviews: Learning the Craft of Qualitative Research Interviewing*. Sage Publications, Inc.
- Lakoff, G., & Johnson, M. (2008). *Metaphors We Live By*. University of Chicago Press.
- Lankhorst, M. M. (2013). *Enterprise Architecture at Work - Modelling, Communication and Analysis*. Springer.
- Lankhorst, M., H.A., P., & Jonkers, H. (2010). The Anatomy of the ArchiMate Language. *International Journal of Information System Modeling and Design*, 1(1), 1-32.
- Lankhorst, M., Proper, H., & Jonkers, H. (2009). The Architecture of the ArchiMate Language. *Proceedings of BPMDS and EMMSAD*. 29. Springer.
- Larkin, J., & Simon, H. (1987). Why a Diagram Is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*, 11(1), 65–100.
- Larman, C. (2005). *Applying UML and Patterns*. Upper Saddle River, New Jersey, U.S.A.: Prentice Hall PTR.
- Lawson, H. (2010). *A Journey Through the Systems Landscape*. Individual author and College Publications.
- Lemoigne, J. L. (1994). *Le Constructivisme*. ESF Editeur.
- Lemos, N. (1999). Value Theory. *The Cambridge Dictionary of Philosophy*.
- Lethbridge, T. M. (2011). Teaching UML Using Umple: Applying Model-Oriented Programming in the Classroom. *Conference on Software Engineering Education and Training*. IEEE.
- Liu, K., Dietz, J. L., & Barjis, J. (2003). *Capturing Organizational Behavior with Dynamic Modeling - Extension of DEMO from a Semiotic Perspective*.
- Lockerbie, J., Bush, D., Maiden, N., Blom, H., & Everdij, M. (2010). Using i* Modelling as a Bridge between Air Traffic Management Operational Concepts and Agent-Based Simulation Analysis. *18th IEEE International Requirements Engineering Conference*. IEEE Computer Society.
- Lohse, G. (1993). A Cognitive Model for Understanding Graphical Perception. 8(4), 353–388.
- Lohse, G. (1997). The Role of Working Memory in Graphical Information Processing. *Behaviour and Information Technology*, 297–308.
- Longépé, C. (2003). *The Enterprise Architecture IT Project: The Urbanisation Paradigm*. Kogan Page Science.
- Lusch, R. F., Vargo, S. L., & Wessels, G. (2008). Toward a Conceptual Foundation for Service Science: Contributions from Service-Dominant Logic. *IBM Systems Journal*, 47(1).
- MacLean, A., Bellotti, V., & Shum, S. (1993). Developing the Design Space with Design Space Analysis. *Computers, Communication and Usability: Design issues, research and methods for integrated services*, 197-219.
- MacLean, A., Young, R. M., Bellotti, V. M., & Moran, T. P. (1991). Questions, Options, and Criteria: Elements of Design Space Analysis. *Human-Computer Interaction*, 6, 201-250.
- Madsen, K. H. (1994). A Guide to Metaphorical Design. *Communications of the ACM, Visualization and Design*, 37(12).
- Maglio, P. P., Vargo, S. L., Caswell, N., & Spohrer, J. (2009, 1 6). The Service System Is the Basic Abstraction of Service Science. *Information Systems E-Business Management*, 7, pp. 395-406.

- March, S. T., & Smith, G. F. (1995). Design and Natural Science Research on Information Technology. *Decision Support Systems*, 15(4), 251–266.
- Markus, L. M., & Keil, M. (1994). If We Build It, They Will Come: Designing Information Systems that People Want to Use. *Sloan Management Review*, 35(4), pp. 11-26.
- Masri, K. P. (2008). Using Iconic Graphics in Entity Relationship Diagrams: The Impact on Understanding. *Journal of Database Management*, 19(3), 22-41.
- Maturana, H. R., & Varela, F. J. (1987). *The Three of Knowledge - The Biological Roots of Human Understanding*. Boston and London: Shambhala Publications.
- Mc Dougall, S., Curry, M., & Bruijn, O. (2001). The Effects of Visual Information on Users' Mental Models: An Evaluation of Pathfinder Analysis as a Measure of Icon Usability. *Cognitive Ergonomics*, 5, 59–84.
- McCabe, A., & Peterson, C. (1984). What Makes a Good Story. *Psycholinguistic Research*, 13(6), 457-480.
- McCloud, S. (1994). *Understanding Comics - The Invisible Art*. Harper Collins Publishers.
- McCloud, S. (2000). *Reinventing Comics - How Imagination and Technology are Revolutionizing an Art Form*. Harper Collins Publishers.
- McCloud, S. (2006). *Making Comics - Storytelling Secrets of Comics, Manga and Graphic Novels*. Harper Collins Publishers.
- Mendling, J., Reijers, H., & van der Aalst, V. (2009). *Seven Process Modeling Guidelines (7PMG)*. Elsevier.
- Merriam-Webster. (2016). *Merriam-Webster Online Dictionary*. Retrieved from <http://www.merriam-webster.com/dictionary/paradigm>
- Miller, G. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review*, 81–97.
- Miller, J. G. (1995). *Living Systems*. University Press of Colorado.
- Mintzberg, H., Ahlstrand, B., & J., L. (1998). *Strategy Safari – The Complete Guide Through the Wilds of Strategic Management*. Prentice Hall.
- Monk, A., & Howard, S. (1998). *The Rich Picture: A Tool for Reasoning about Work Context. Methods and Tools*.
- Moody, D. (2009). Theory Development in Visual Language Research: Beyond the Cognitive Dimensions of Notations. *Visual Languages And Human-Centric Computing*, 151–154.
- Moody, D. L. (2009). The "Physics" of Notations: Towards a Scientific Basis for Constructing Visual Notations in Software Engineering. *Transactions on Software Engineering*, 35(5), 756-777.
- Moody, D. L. (2009). The Physics of Notations: Improving the Usability and Communicability of Visual Notations in Requirements Engineering. *International Workshop on Requirements Engineering Visualization*. Atlanta: IEEE Computer Society.
- Moody, D. L. (2010). The Physics of Notations: A Scientific Approach to Designing Visual Notations for Requirements Engineering. *Requirements Engineering Conference*. Atlanta.
- Moody, D. L., Genon, N., Heymans, P., & Caire, P. (n.d.). *Visual Notation Design 2.0: Designing User-Comprehensible Diagramming Notations*.
- Moody, D., & al., e. (2002). Evaluating the Quality of Process Models: Empirical Testing of a Quality Framework. *Proceedings LNCS*, 2503, pp. 380-396.

- Moody, D., & van Hillegersberg, J. (2008). Evaluating the Visual Syntax of UML: An Analysis of the Cognitive Effectiveness of the UML Family of Diagrams. *International Conference On Software Language Engineering*. Springer-Verlag.
- Moody, D., Heymans, P., & Matulevicius, R. (2009). Improving The Effectiveness of Visual Representations in Requirements Engineering: An Evaluation of the i* Visual Notation. *International Conference on Requirements Engineering*. Atlanta: IEEE Computer Society.
- Moody, D., Heymans, P., & Matulevicius, R. (2010). Visual Syntax Does Matter: Improving the Cognitive Effectiveness of the i* Visual Notation. *Requirements Engineering*, 15(2), 141–175.
- Morris, S., & Gotel, O. (2006). Flow Diagrams: Rise and Fall of the First Software Engineering Notation. *Diagrams*.
- Mortensen, K. (2009). *Four Essential Phases of a Great Story*. Retrieved 4 4, 2016, from <http://short.fictionfactor.com/articles/phases.html>
- Mussbacher, G. (2011). *The Aspect-Oriented User Requirements Notation (AoURN): Aspects, Gols, and Scenarios*. University of Ottawa.
- Mussbacher, G., Amyot, D., Araujo, J., & Moreira, A. (2011). *Requirements Modeling with the Apect-Oriented User Requirements Notation (AoURN): A Case Study*. University of Ottawa.
- Mylopoulos, J., Chung, L., & Nixon, B. (1992). Representing and Using Non-Functional Requirements: A Process-oriented Approach. *Transactions on Software Engineering*. IEEE.
- Mylopoulos, J., Chung, L., & Yu, E. (1999, 01). From Object-Oriented to Goal-Oriented Requirements Analysis. *Communications of the ACM*, 42(1), pp. 31-37.
- Mylopoulos, J., Chung, L., Liao, S., & Wang, H. (2001). Exploring Alternatives during Requirements Analysis. *Requirements Engineering*.
- Narayanan, H. N., & Hübscher, R. (1997). *Visual Language Theory: Towards a Human-Computer Interaction Perspective*. University of Virginia.
- Naumenko, A., & Wegmann, A. (2002). *Triune Continuum Paradigm: A Paradigm for General System Modeling and Its Applications for UML and RM-ODP*. Ecole Polytechnique Fédérale de Lausanne. Lausanne: EPFL.
- Networked Business Models with e3-value*. (2016, 4 4). Retrieved from e3-value: <http://e3value.few.vu.nl/>
- Nonaka, I., & Takeuchi, H. (1995). *The Knowledge-Creating Company - Jow Japanese Companies Create the Dynamics of Innovation*.
- Nonaka, I., Toyama, R., & Konno, N. (2000). SECI, Ba and Leadership: A Unified Model of Dynamic Knowledge Creation. 33(1), 5-34.
- Norman, D. A. (1990). *The Design of Everyday Things*. New York, New York, U.S.A.: Doubleday.
- Novick, L. (2006). The Importance of Both Diagrammatic Conventions and Domain-Specific Knowledge for Diagram Literacy in Science: The Hierarchy as an Illustrative Case in Diagrammatic Representation and Inference. Berlin: Springer.
- Object-Management-Group. (2016, 08 29). OMG Model Driven Architecture. Retrieved 08 29, 2016, from Object Management Group: <http://www.omg.org/mda/>
- Osterwalder, A., & Pigneur, Y. (2016, 07 20). *Business Model Generation*. Retrieved from <http://www.businessmodelgeneration.com/>
- Paivio, A. (1986). *Mental Representations: A Dual Coding Approach*. Oxford, U.K.: Oxford University Press.

- Park, A. (2016, 07 05). *The RSA*. Retrieved from RS Animate: <https://www.thersa.org/discover/videos/rsa-animate>
- Peltier, J. (2011, 11). Luxury Coach. *SEAM Modeling Workshop – Business / IT Strategy*. Lausanne, Vaud, Switzerland: 3rd SEAM Adopter Meeting.
- Perroud, O., Wegmann, A., & Montserrat, J. (2005). *Evaluation of the SEAM in Business Methodology*. Ecole Polytechnique Fédérale de Lausanne, Management of Technology, Lausanne.
- Petre, M. (1995). Why Looking Isn't Always Seeing: Readership Skills and Graphical Programming. *Communications of ACM*, 38(6), 33–44.
- Polanyi, M. (1983). *The Tacit Dimension*. Doubleday & Company, Inc.
- Popescu, G., & Wegmann, A. (2014). Using The Physics of Notations Theory to Evaluate the Visual Notation of SEAM. *16th Conference on Business Informatics (CBI)*. Geneva: IEEE.
- Popescu, G., Tapandjieva, G., & Wegmann, A. (2013). Business and IT Design with SEAM: An Illustration with the PhD Hiring Process at École Polytechnique Fédérale de Lausanne. *IEEE International Conference on Systems, Man and Cybernetics (SMC)*. Manchester.
- Porter, M. E. (1980). *Competitive Strategy: Techniques for Analyzing Industries and Competitors*. New York, U.S.A.: The Free Press.
- Regev, G. (2003). *The Lightswitch Approach - A Systemic Paradigm for Early IT System Requirements Based on Regulation Principles*. Lausanne, Vaud, Switzerland.
- Regev, G. (n.d.). *A Systemic Paradigm for Early IT System Requirements Based on Regulation Principles*. Ecole Polytechnique Fédérale de Lausanne, Lausanne.
- Regev, G., & Wegmann, A. (2004). Defining Early IT System Requirements with Regulation Principles: The Lightswitch Approach. *Proceedings of the 12th IEEE International Requirements Engineering Conference* (pp. 144–153). IEEE.
- Regev, G., & Wegmann, A. (2005). Where Do Goals Come From: The Underlying Principles of Goal-Oriented Requirements Engineering. *Requirements Engineering Conference*. Paris.
- Regev, G., & Wegmann, A. (n.d.). Regulation, the Invisible Part of the Goal Oriented Requirements Engineering Iceberg. *International Symposium on Business Modeling and Software Design*, (pp. 42-50). 2011.
- Regev, G., Bajic-Bizumic, B., Golnam, A., Popescu, G., Tapandjieva, G., Saxena, A. B., & Wegmann, A. (2013). A Philosophical Foundation for Business And IT Alignment in Enterprise Architecture with the Example of SEAM. *Proceedings of Business Modeling and Software Design* (pp. 131-139). Noordwijkerhout: Springer.
- Regev, G., Gause, D. C., & Wegmann, A. (2009, 12). Experiential Learning Approach for Requirements Engineering Education. *Requirements Engineering*, 14(4), 269-287.
- Regev, G., Hayard, O., & Wegmann, A. (2011). Service Systems and Value Modeling from an Appreciative System Perspective. *Exploring Services Science*, 82, 146-157.
- Rittel, H. (2016). *IBIS: A Tool for All Reasons*. Retrieved from CogNexus: http://www.cognexus.org/IBIS-A_Tool_for_All_Reasons.pdf
- Rittel, H. W., & Webber, M. M. (1973). Dilemmas in a General Theory of Planning. *Policy Sciences*(4), pp. 155-169.
- Robertson, S., & Robertson, J. (2008). *Volere Requirements Techniques: An Overview*. The Atlantic Systems Guild Ltd.
- Rollanda, C., Nurcana, S., & Grosza, G. (1999). Enterprise Knowledge Development: The Process View. *Information and Management*, 36(3), 165–184.

- Rumbaugh, J. (1991). *Object-Oriented Modeling and Design*. Englewood Cliffs, NJ, U.S.A.: Prentice-Hall.
- Rychkova, I., Regev, G., Lê, L.-S., & Wegmann, A. (2007). From Business to IT with SEAM: J2EE Pet Store Example. *International EDOC Conference*. Annapolis: IEEE.
- Sankaran, S., Hou, T. B., & Orr, M. (2008). Incorporating Systems Thinking in Organizational Change Projects Using Action Research by Practitioners Conducting Academic Research. *International Society for the Systems Sciences*.
- Sassoon, J. (1998). *Urbanisation des Systèmes D'information*. Hermès.
- Schuette, R. R. (1998). The Guidelines of Modeling - An Approach to Enhance the Quality in Information Models. *Lecture Notes in Computer Science, 1507*, 240–254.
- Shaft, T. V. (2006). The Role of Cognitive Fit in the Relationship between Software Comprehension and Modification. *MIS Quarterly, 30*(1), 29–55.
- Shekerman, J. (2004). *How to Survive in the Jungle of Enterprise Architecture Frameworks*.
- Shingo, S. (1986, 4 1). *Zero Quality Control: Source Inspection and the Poka-Yoke System*. Portland: Productivity Press.
- Shingo, S. (n.d.). *The Sayings of Shigeo Shingo: Key Strategies for Plant Improvement*. Productivity Press.
- Shum, S., MacLean, A., Bellotti, V., & Hammond, N. (1996). Graphical Argumentation and Design Cognition. *Knowledge Media Institute*.
- Siau, K. (2004). Informational and Computational Equivalence in Comparing Information Modelling Methods. *Journal of Database Management, 15*(1), 73–86.
- Siau, K., & Cao, Q. (2001). Unified Modeling Language: A Complexity Analysis. *Journal of Database Management, 12*, 26-34.
- Simon, H. A. (1969). *The Sciences of the Artificial*. Cambridge, USA: MIT Press.
- Snene, M., Ralyte, J., Morin, J.-H., Golnam, A., Regev, G., Ramboz, J., . . . Wegmann, A. (2010). Systemic Service Design: Aligning Value and Implementation. *International Conference on Exploring Services Sciences*, (pp. 150-164).
- Soccard, A. T. (2016). *Interpretations of Service Models - Visualisation and Representation of Models*. EPFL, Computer Science. Lausanne: EPFL.
- Sowa, J. F. (2000). *Knowledge Representation. Logical, Philosophical and Computational Foundations*. (C. Brooks, Ed.) MIT Press.
- Sowa, J. F., & Zachman, J. A. (1992). Extending and Formalizing the Framework for Information Systems Architecture. *IBM Systems Journal, 31*(3), 590-616.
- Spohrer, J., Maglio, P. P., Bailey, J., & Gruhl, D. (2007, 1). Steps Toward a Science of Service Systems. *IBM Almaden Research Center*.
- Storyjumper. (2010). *Workbook - Learn to Tell a Story in Just 7 Steps*. Retrieved 4 4, 2016, from Storystarter: <https://www.storyjumper.com/files/storystarter.pdf>
- Strauss, A. L., & Corbin, J. M. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Sage Publications, Inc.
- The Open Group*. (2016, 07 28). Retrieved from ArchiMate notation: http://pubs.opengroup.org/architecture/archimate-doc/ts_archimate/ts_archimate_files/image161.png

- The Open Group Architecture Framework.* (n.d.). Retrieved from TOGAF Version 9.1: <http://www.opengroup.org/togaf/>
- The Pop-Up Generation. Design between Dimensions.* (n.d.). Retrieved from <http://www.mudac.ch/exhibitions/publications/catalogue-pop-up/>
- Thomas, J. C., Diament, J., Martino, J., & Bellamy, R. K. (2012). Using the “Physics” of Notations to Analyze a Visual Representation of Business Decision Modeling. *Visual Languages and Human-Centric Computing*. IEEE.
- Toulmin, S. (1958). *The Uses of Argument*. Cambridge: Cambridge University Press.
- Travis, C. (2001). *Unshadowed Thought: Representation in Thought and Language*. Harvard University Press.
- Trochim, W. M. (2016). *Research Methods Knowledge Base*.
- Tufte, E. R. (1990). *Envisioning Information*. Cheshire: Graphics Press.
- Turetken, O., Schuff, D., Sharda, R., & Ow, T. (2004). Supporting Systems Analysis and Design Through Fisheye Views. *Communications of ACM*, 47(9), 72–77.
- Unified Modeling Language.* (2016, 07). Retrieved from UML diagrams: <http://www.uml-diagrams.org/uml-25-diagrams.html>
- Unified Modeling Language.* (2016). Retrieved from Object Management Group: <http://www.omg.org/>
- URBA-SI, C. (2003). *Pratiques de l'Urbanisme des Systèmes d'Information en Entreprises*. Publibook.
- van der Linden, D., & Hadar, I. (2016). Evaluating the Evaluators - An Analysis of Cognitive Effectiveness Efforts for Visual Notations. *Evaluation Of Novel Approaches To Software Engineering*. ScitePress.
- van der Linden, D., & Hadar, I. (2016, 06 07). User Involvement in Applications of the PoN. *Advanced Information Systems Engineering Workshops*, 249, pp. 109-115.
- van der Linden, D., Zamansky, A., & Hadar, I. (2016). How Cognitively Effective Is a Visual Notation? On the Inherent Difficulty of Operationalizing the Physics of Notations. *Exploring Modeling Methods For Systems Analysis And Design*. Springer.
- van Lamsweerde, A., Darimont, R., & Massonet, P. (1995, 03). Goal Directed Elaboration of Requirements for a Meeting Scheduler: Problems and Lessons Learnt. *IEEE International Symposium on Requirements Engineering*, pp. 194-203.
- van Reijswoud, V. E., Mulder, H. B., & Dietz, J. L. (2001, 12 25). Communicative Action-Based Business Process and Information Systems Modelling with DEMO. *Information Systems Journal*, 9(2), 117-138.
- Vargo, S. L., & Lusch, R. F. (2006). Service-Dominant Logic: What It Is, What It Is Not, What It Might Be. In S. L. Vargo, *The Service-dominant Logic Of Marketing: Dialog, Debate, And Directions*. (pp. 43–56).
- Vargo, S., Maglio, P. P., & Akaka, M. A. (2008). On Value and Value Co-Creation: A Service Systems and Service Logic Perspective. *European Management Journal*, 26(3), 145–152.
- Vessey, I. (1991). Cognitive Fit: A Theory-Based Analysis of the Graphs versus Tables Literature. *Decision Sciences*, 22(2), 219-240.
- Vickers, G. S. (1968). *Value Systems and Social Process*. London, U.K.: Tavistock Press.
- von Bertalanffy, L. (1968). *General System Theory*. New York: George Braziller.
- Vygotsky, L. (1997). *Thought and Language*. Cambridge, MA, USA: MIT Press.

- Wand, Y., & Weber, R. (1990). An Ontological Model of an Information System. *Transactions on Software Engineering*, 1282-1292.
- Weber, R. (1997). *Ontological Foundations of Information Systems*. Coopers And Lybrand.
- Wegmann, A. (2003). On the Systemic Enterprise Architecture Methodology (SEAM). *International Conference on Enterprise Information Systems*, (pp. 483-490).
- Wegmann, A. (2012). Modeling Business Strategy for SITRA. Lausanne, Vaud, Switzerland.
- Wegmann, A. (2016, 4 4). *Systemic Enterprise Architecture Method*. Retrieved 4 4, 2016, from Laboratory of Systemic Modeling, EPFL: <http://lams.epfl.ch/reference/seam/>
- Wegmann, A., & Genilloud, G. (2000). *The Role of "Roles" in Use Case Diagrams*. Technical report, Ecole Polytechnique Fédérale de Lausanne, Department of Computer Science, Lausanne.
- Wegmann, A., & Rey, D. A. (2016). Modeling the Information Systems Strategy for EPFL. Lausanne, Vaud, Switzerland.
- Wegmann, A., Bajic-Bizumic, B., Golnam, A., Popescu, G., Tapandjieva, G., Saxena, A. B., . . . Regev, G. (2013). Requirements Modeling in SEAM: The Example of a Car Crash Management System. *Proceedings Of Requirements Engineering, Comparing Requirements Modeling Approaches Workshop* (pp. 25-30). Rio de Janeiro: Springer.
- Wegmann, A., Regev, G., Rychkova, I., Julia, P., & Perroud, O. (2007). Early Requirements and Business-IT Alignment with SEAM for Business. *15th International Requirements Engineering Conference*. New Delhi: IEEE.
- Wegmann, A., Regev, G., Rychkova, I., Lê, L.-S., & De La Cruz, J. D. (2007). Business-IT Alignment with SEAM for Enterprise Architecture. *International EDOC Conference*. Annapolis: IEEE.
- Wegmann, A., Regev, G., Viotti, G., Pilet, P.-A., & Vuagniaux, P. (2012). Modeling an Emergency Service System for a Hospital Network. *International Conference on Exploring Service Science*. Geneva.
- Wegmann, P. (2000). Kinetic Family Drawing Scoring Method for Cross-Cultural Studies. *The Arts in Psychotherapy*, 27(3), 179-190.
- Weick, K. E. (1979). *The Social Psychology of Organizing*. Random House.
- Weinberg, G. M. (1975). *An Introduction to General Systems Thinking*. New York: Dorset House Publishing Co.
- Weinberg, G. M. (1985). *The Secrets of Consulting - A Guide to Giving and Getting Advice Successfully*. New York, New York, U.S.A.: Dorset House Publishing.
- Weinberg, G. M., & Weinberg, D. (1988). *General Principles of System Design*. Dorset House.
- Whetten, D. A. (1989). What Constitutes a Theoretical Contribution. *Academy of Management Review*, 14(4), 490-495.
- White, A. (n.d.). *The Elements of Graphic Design: Space, Unity*. 2002.
- Wilson, B. (1990). *Systems: Concepts, Methodologies, and Applications*.
- Winn, W. (1993). An Account of How Readers Search for Information in Diagrams. *Contemporary Educational Psychology*, 18, 162-185.
- Winograd, T., & Flores, F. (1986). *Understanding Computers and Cognition - A New Foundation for Design*. Addison Wesley Longman, Inc.

- Woods, E., & Bashroush, R. (2015). Modeling Large-Scale Information Systems Using ADLS - An Industrial Experience Report. *Journal of Systems and Software*, 99, 97-108.
- Yin, R. K. (1994). *Case Study Research: Design and Methods* (Vol. 5). (A. S. Series, Ed.) London, U.K.: Sage Publications.
- Yu, E. (1995). *Modelling Strategic Relationships for Process Reengineering*. University of Toronto, Graduate Department of Computer Science. Toronto: University of Toronto.
- Yu, E. (2001, December 13-14). Strategic Actor Relationships Modelling with i*. Trento: IRST.
- Yu, E. (2001, 12 13). *University of Toronto*. Retrieved 4 5, 2016, from Publications: <http://www.cs.utoronto.ca/pub/eric/tut1.1-v2.ppt>
- Yu, E. S. (1997). Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. *Requirements Engineering* (pp. 226-235). IEEE Computer Society.
- Yu, E. S., & Mylopoulos, J. (1994). From E-R to "A-R" - Modelling Strategic Actor Relationships for Business Process Reengineering. *Business Modelling and Re-Engineering Lecture Notes in Computer Science*, 881, 548-565.
- Zachman, J. A. (1987). A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3), 276-292.
- Zachman, J. A. (2004). The Zachman Framework and Observations on Methodologies. *Business Rules Journal*, 5(11).
- Zelm, M., Vernadat, F. B., & Kosanke, K. (1995). The CIMOSA Business Modelling Process. *Computers in Industry*, 27, 123-142.
- Zhang, J. (n.d.). The Nature of External Representations in Problem Solving. *Cognitive Science*, 21(2), 179-217.

Appendix 1 Case of Car-Maintenance Service with SEAM

We include other model iterations for the case of car-maintenance service modeled with the SEAM notation. They are useful to compare against the proposed model iterations presented in Chapter 4.5.3 in terms of choice of graphical elements and terminology used across model instances.

2nd iteration

Compared with the first model iteration (original SEAM model) presented in Chapter 4.3, the model from Figure 78 includes only one car dealer that is shown both as whole (only with the service and the two properties: price and unavailability time) and as composite (with the “Repair George’s car” process and the five entities that implement this process: a breaks supplier that provides new breaks, a secretary that prepares and sends the bill, a mechanic that diagnoses and repairs the car, an IT system that manages the process, and a salesman that manages sales).

This second iteration and the original (first) model iteration were done using the SeamCAD tool (Carrupt & Wegmann, 2016).

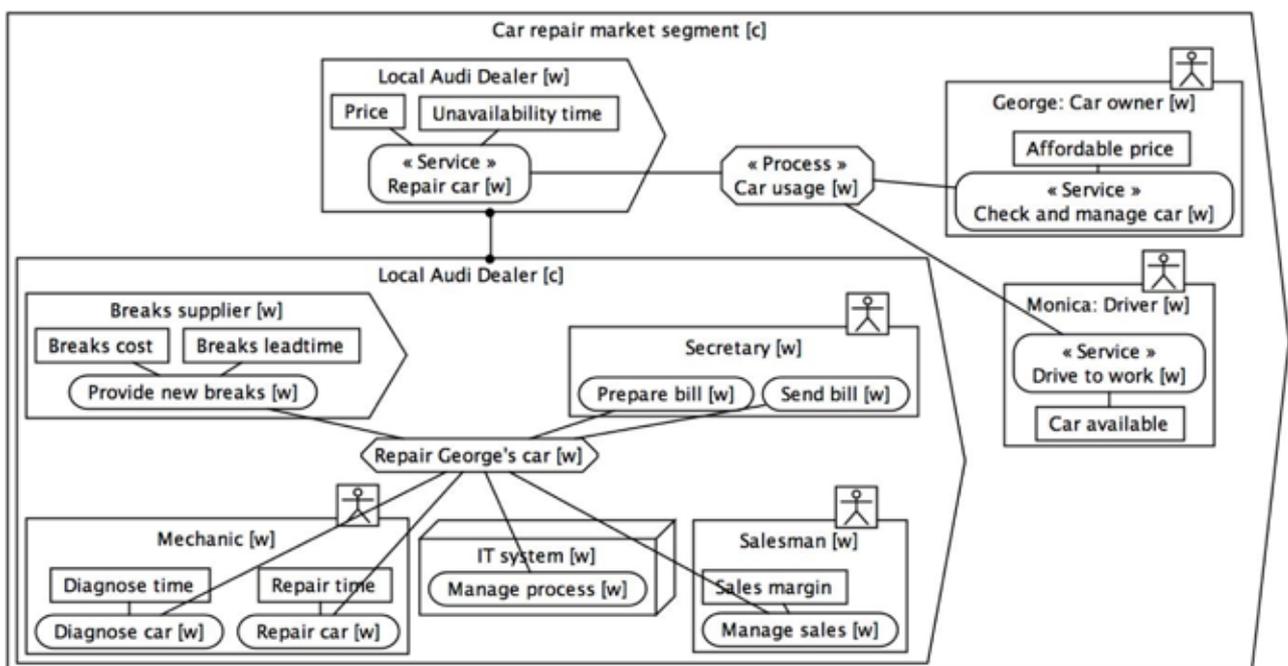


Figure 78: SEAM car-maintenance service model showing the car dealer as whole and as composite

3rd iteration

From the third model iteration, all following models were done in Microsoft PowerPoint, because of the tools flexibility to accommodate various visual elements, their colors and positioning. The third model iteration contains two model instances. First, in Figure 79, the two dealers are shown as wholes. Color-coding is used to mark the relations between the goals of each customer and the service offerings of the suppliers. Then, in Figure 80, only one dealer is shown as a composite. All properties related to price are colored in green and all related to time are in red.

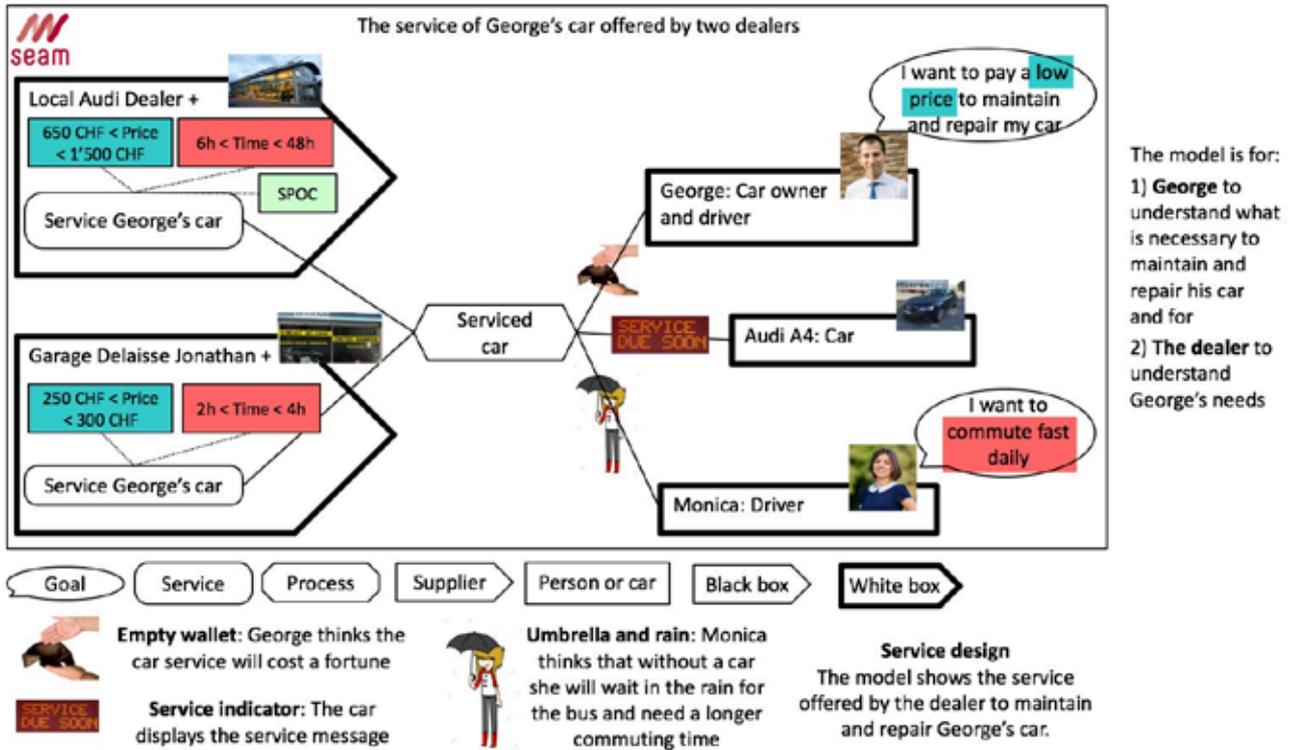


Figure 79: SEAM car-maintenance service model showing the two dealers as wholes

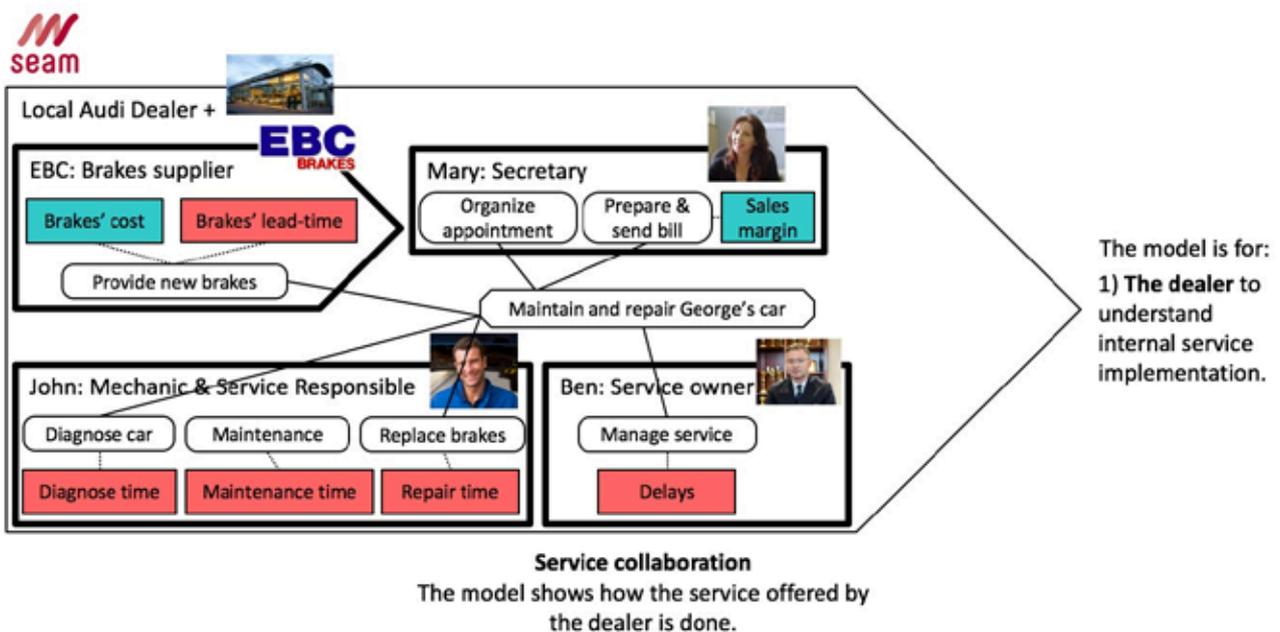


Figure 80: SEAM car-maintenance service model showing one dealer as a composite

4th iteration

Compared with the previous iteration, in Figure 81 and Figure 82 we distinguish visually between wholes (grey background) and composites (white background). George and Monica's goals are stated more explicitly. There is a boundary around them and the car suggesting that they need to decide together as a larger entity. The decision is shown with a set of "thumbs-up" / "thumbs-down" graphical symbols. The generic "Local Audi Dealer" is replaced with "Amag Audi Dealer Geneva". The viewpoint of the modeler is additionally included at the top using the metaphor of binoculars.

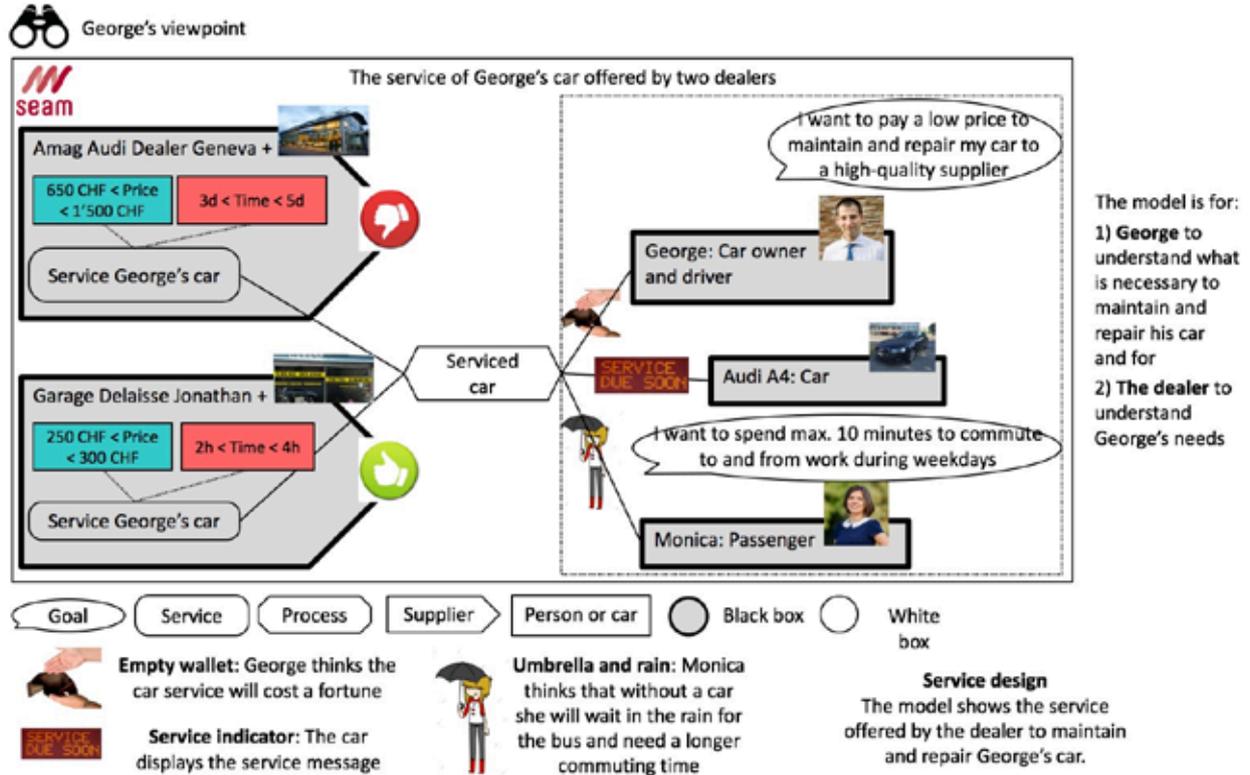


Figure 81: SEAM car-maintenance service model showing wholes and composites more explicitly

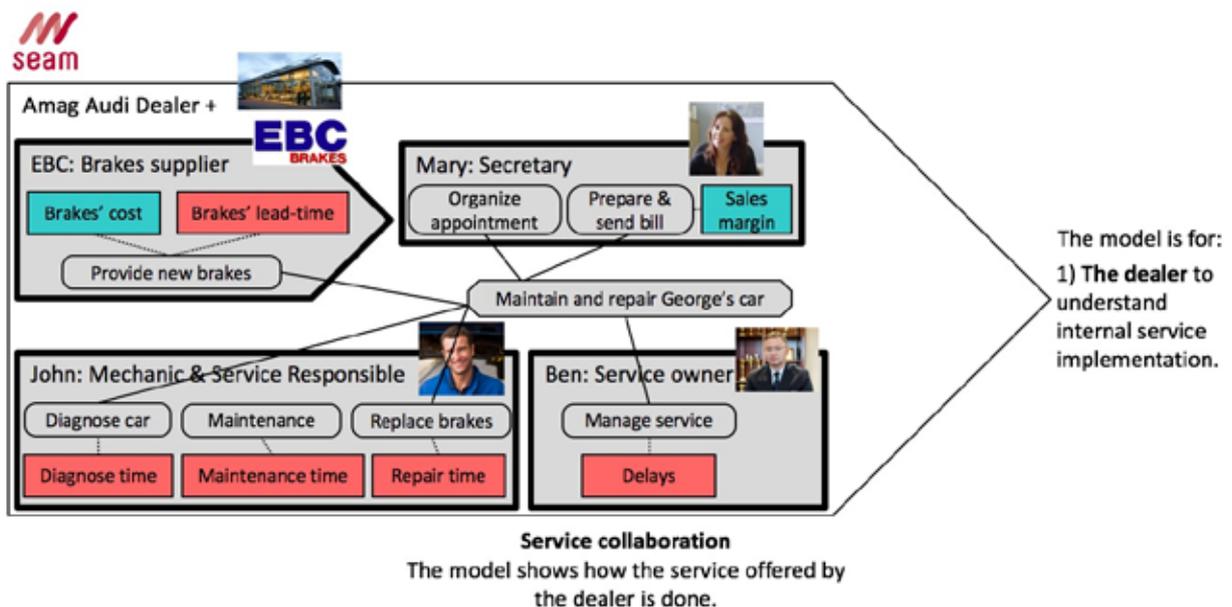


Figure 82: SEAM car-maintenance service model showing the composition of one dealer

5th iteration

With this model iteration we modeled the story more explicitly by using a sequence of model instances that are included in the following figures. We start by modeling the customers to the right by using a house metaphor to suggest that they represent a family. We highlight in bold the keywords for the goals of each customer. The viewpoint is shown on all model instances at the top, in the middle. The two dealers have a distinguishable photo. We added the replacement car for Amag. After the customer's choice, we show the car with Delaisse with its hood open. After the service we show it with the hood closed. The last model instance shows the car with the customers (car owners) and the money with Delaisse (price for the service).

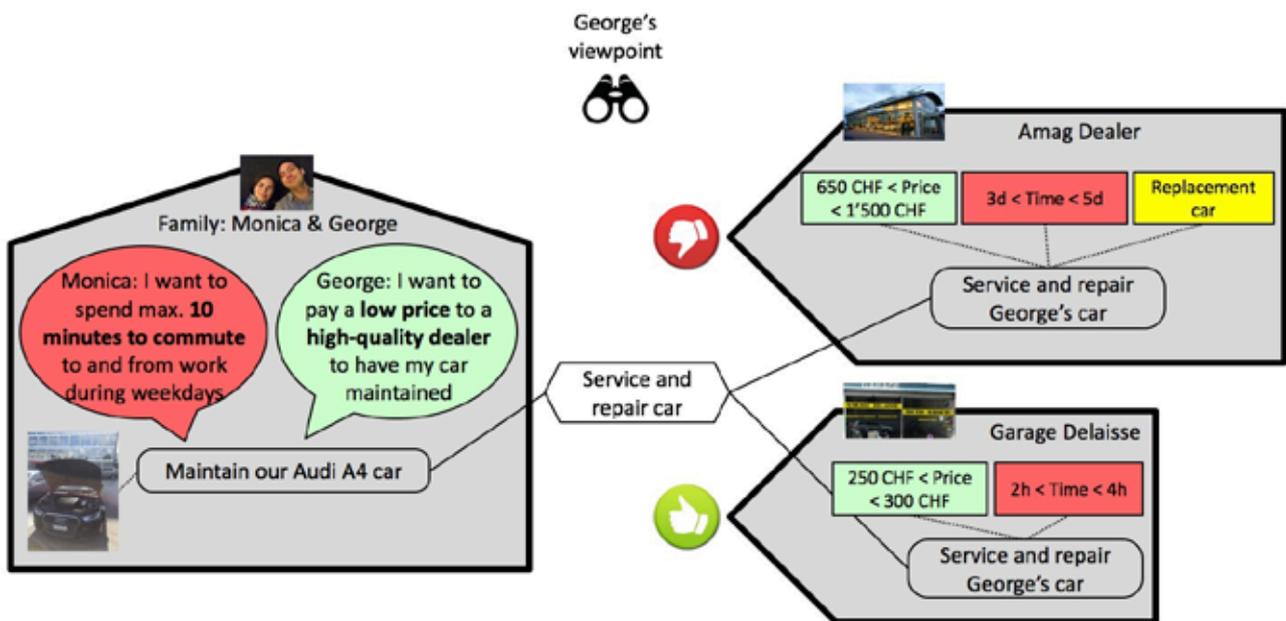


Figure 83: SEAM car-maintenance service model showing the customers to the left and the dealers to the right

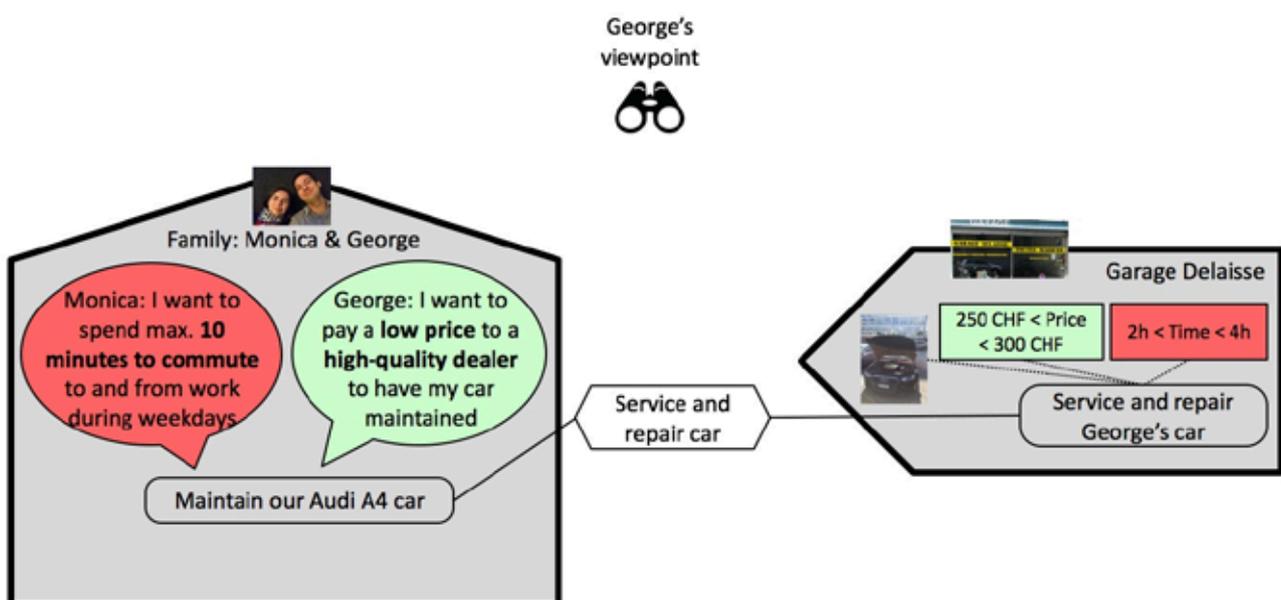


Figure 84: SEAM car-maintenance service model showing the car with the hood open at the dealer during service

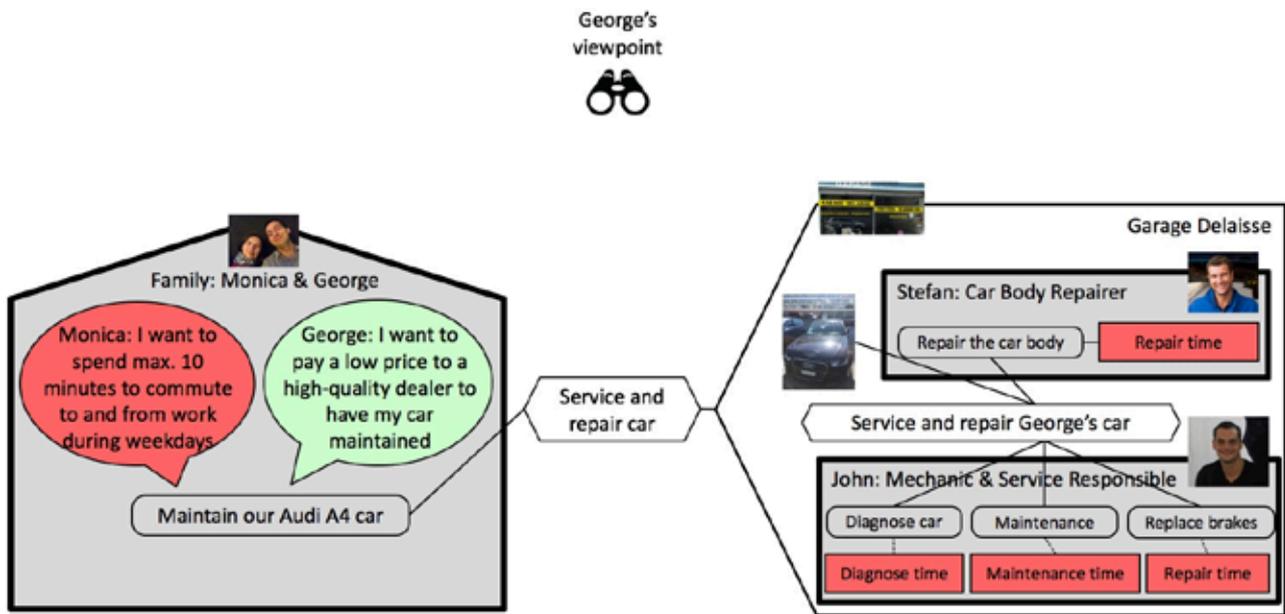


Figure 85: SEAM car-maintenance service model showing the implementation of the service by the dealer

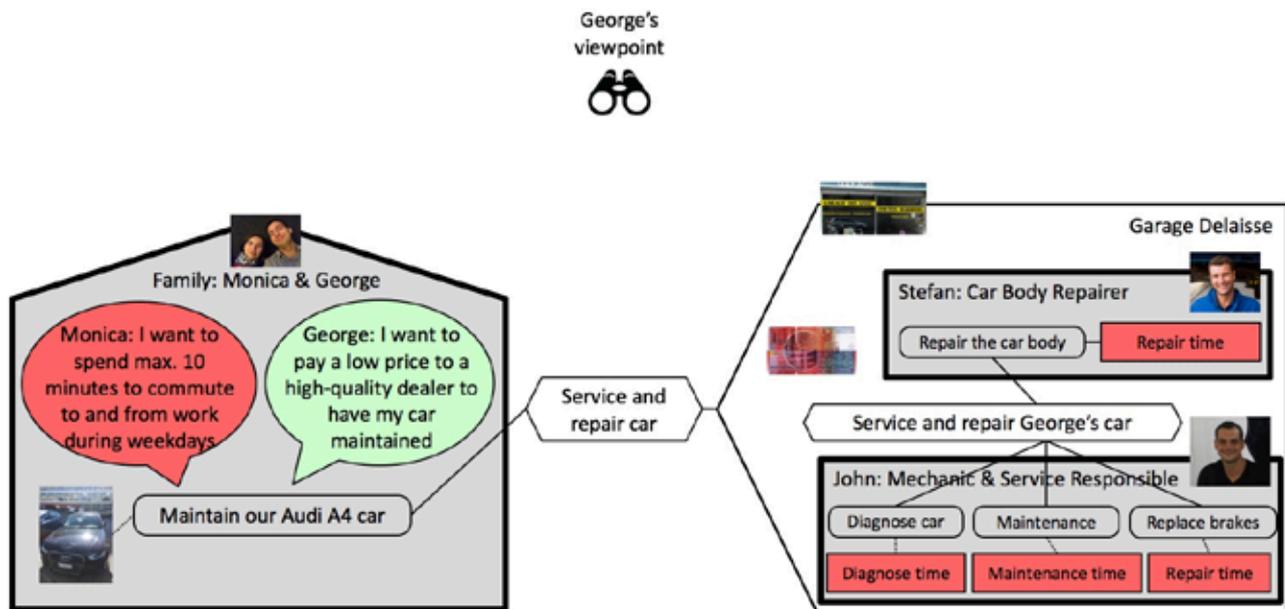


Figure 86: SEAM car-maintenance service model showing the exchange of money for the serviced car

8th iteration

In this iteration we repositioned the customers to the right and the suppliers to the left. We kept the “thumbs-up” graphical symbol for Delaisse after removing the Amag dealer to further suggest the choice for this supplier. We added the “Service due soon” graphical icon to suggest that the car needs to be serviced. In the view of the Delaisse dealer as a composite we added the “Low cost infrastructure” to justify the reduced price.

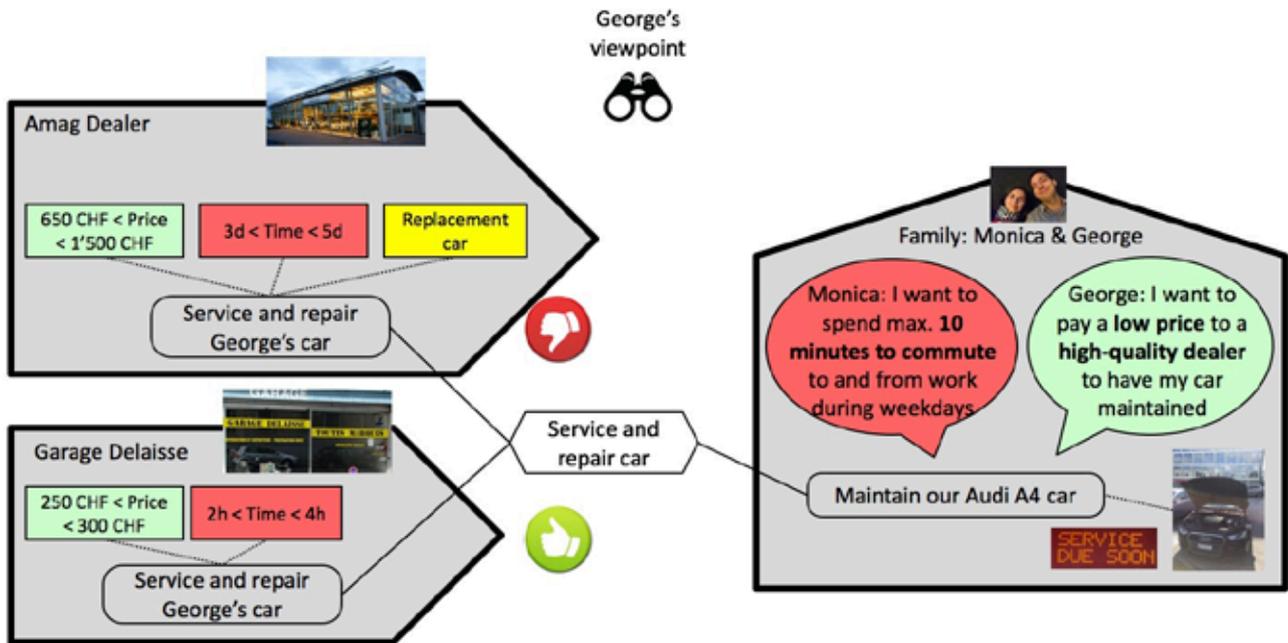


Figure 87: SEAM car-maintenance service model showing back the customers to the right and the suppliers to the left

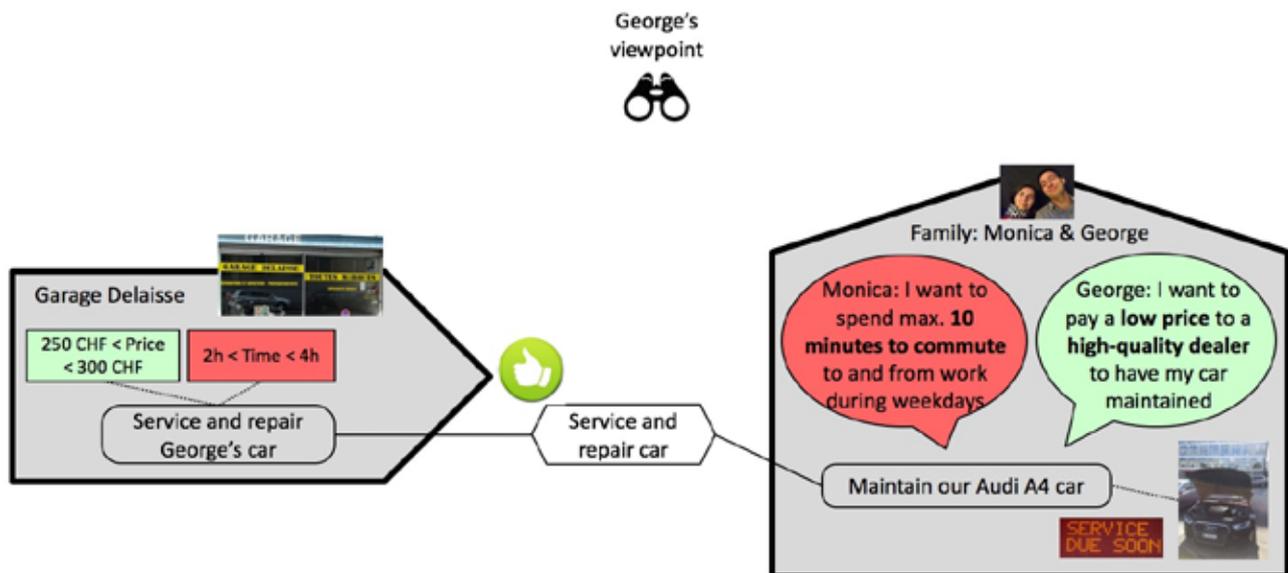


Figure 88: SEAM car-maintenance service model showing the choice confirmation for one dealer

George's
viewpoint

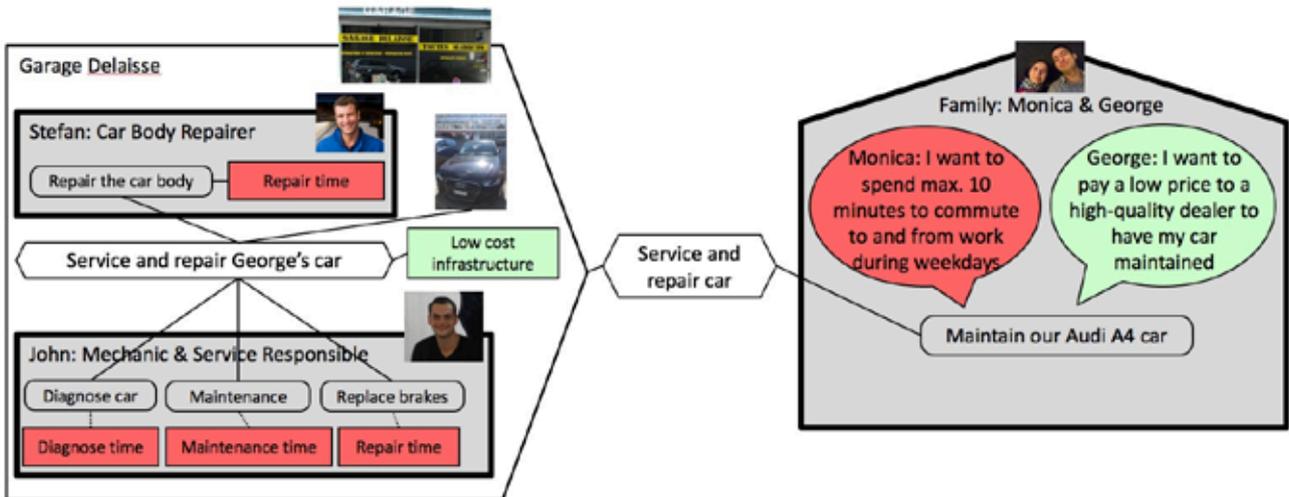



Figure 89: SEAM car-maintenance service model showing the low-cost implementation justified by the infrastructure

George's
viewpoint

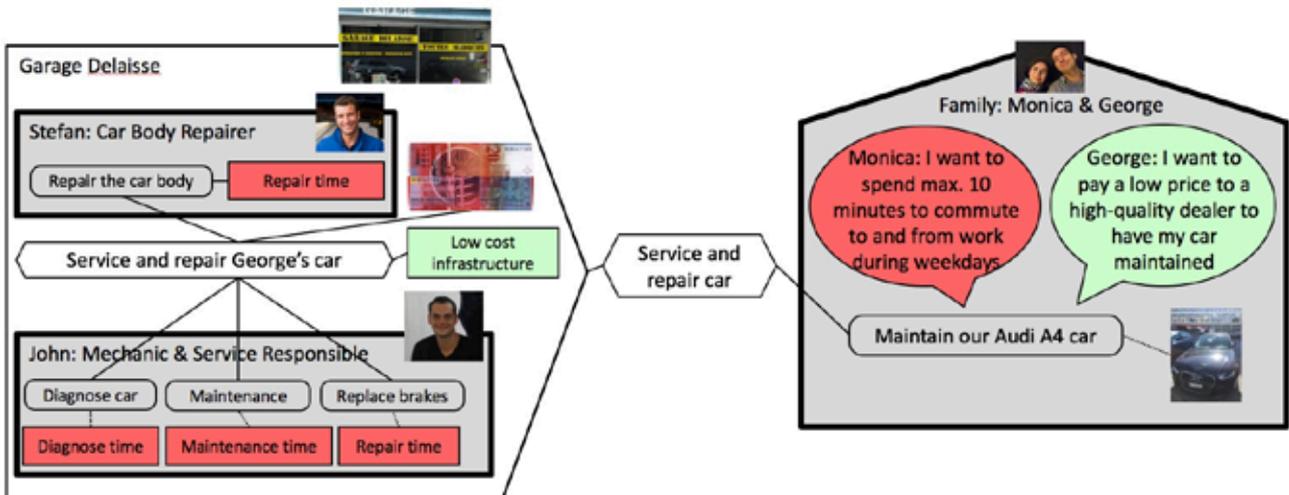



Figure 90: SEAM car-maintenance service model showing the completion of the service with payment

12th iteration

In this iteration we return to the two model instances explanation of the story. In the first figure, we show George's viewpoint whereas in the second, John's viewpoint (the dealer). We start by using the photo of a real house for the customers and the photos of the dealers, use George and Monica's photos separately, show the "service due soon" indicator directly on the car (not separately), split George and Monica's beliefs from their goals, add graphical symbols for the duration, replacement car and price. In addition, show the date when the model was created together with the model's author information at the top right. We add the photos of the points of contact from the two dealers. We summarize the story using the sequence shown at the top right: car with the hood open, dealer, and car with the hood closed. For the implementation, we use the photo from inside the garage.

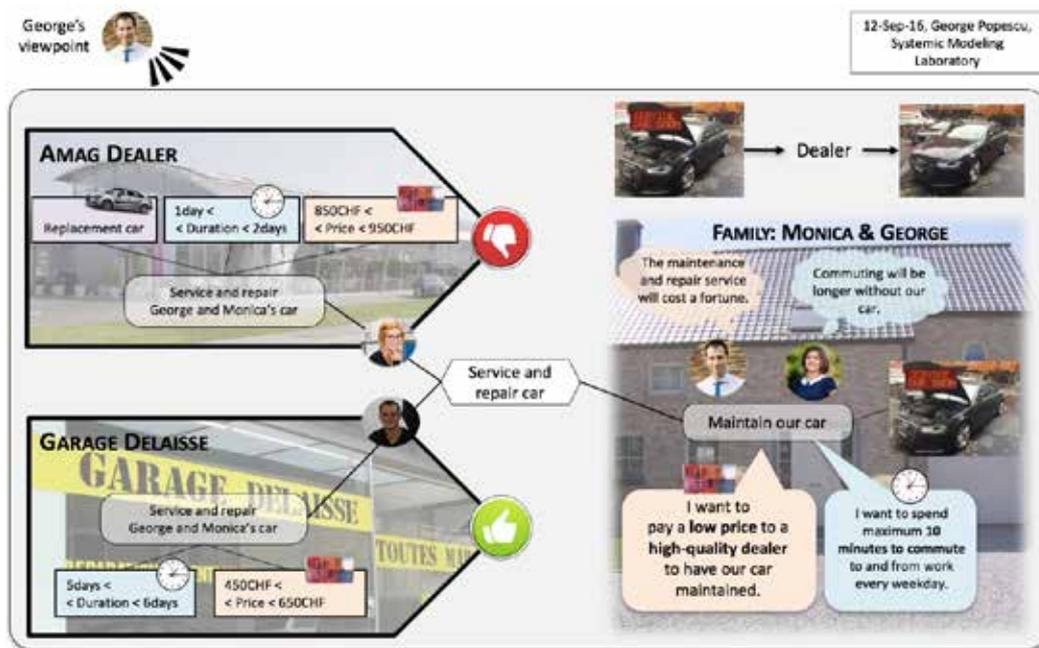


Figure 91: SEAM car-maintenance service model showing the goals and beliefs of the two customers separately

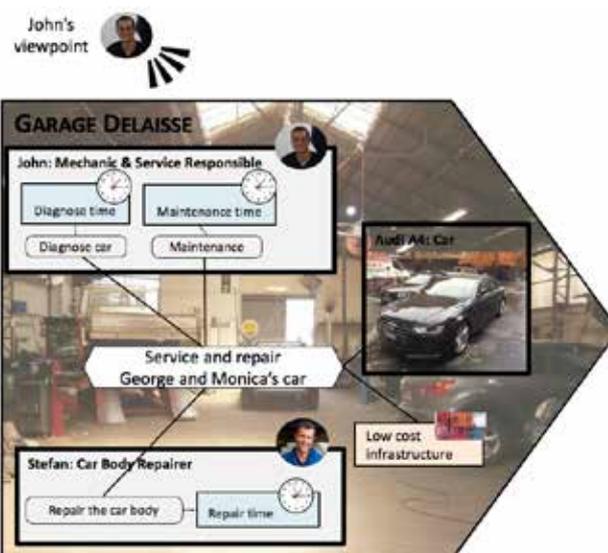


Figure 92: SEAM car-maintenance service model showing the distinction between the dealer shown as whole (photo from the outside) and as composite (photo from the inside)

16th iteration

In this iteration we show two processes (instead of one) in the middle of the model together with a pair of check-marks to suggest the customer's choice. We replace the house metaphor with a photo of George and Monica's flat entrance door. We reformulate the beliefs and goals of George and Monica. We use a different photo of George, at the top left, to suggest his role as a modeler. We investigate different graphical symbols (bikes, bus) to show what happens when the car is in service and the fact that George and Monica need to commute to work with other means of transportation.

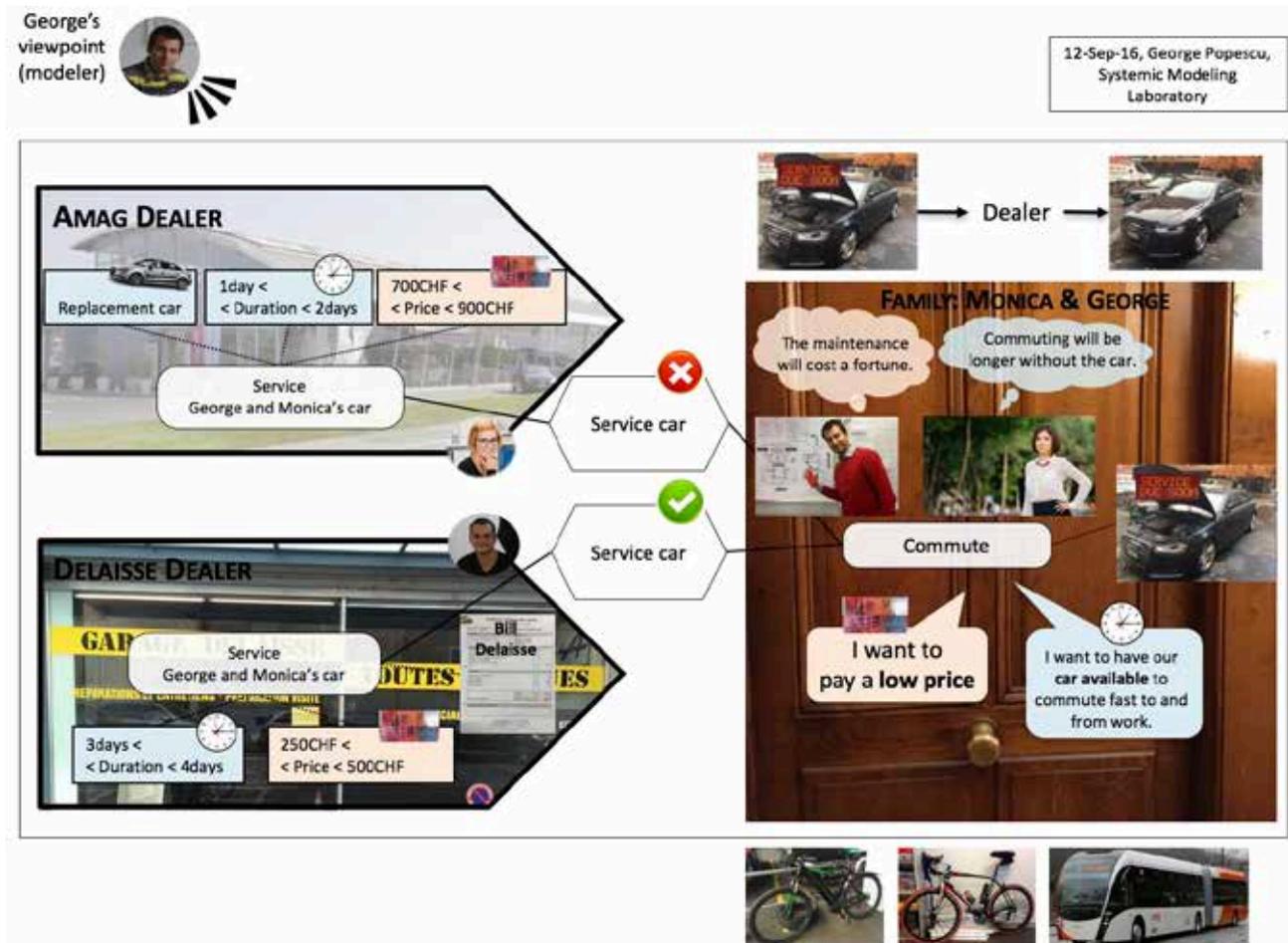


Figure 93: SEAM car-maintenance service model showing the choice of dealers with two processes

18th iteration

Compared with the proposed model presented in Chapter 4.5.3 the model from Figure 94 includes the viewpoints (of the modeler and of actors) for all systems and different graphical metaphors for the time, the money and the replacement car. The main question is not shown.

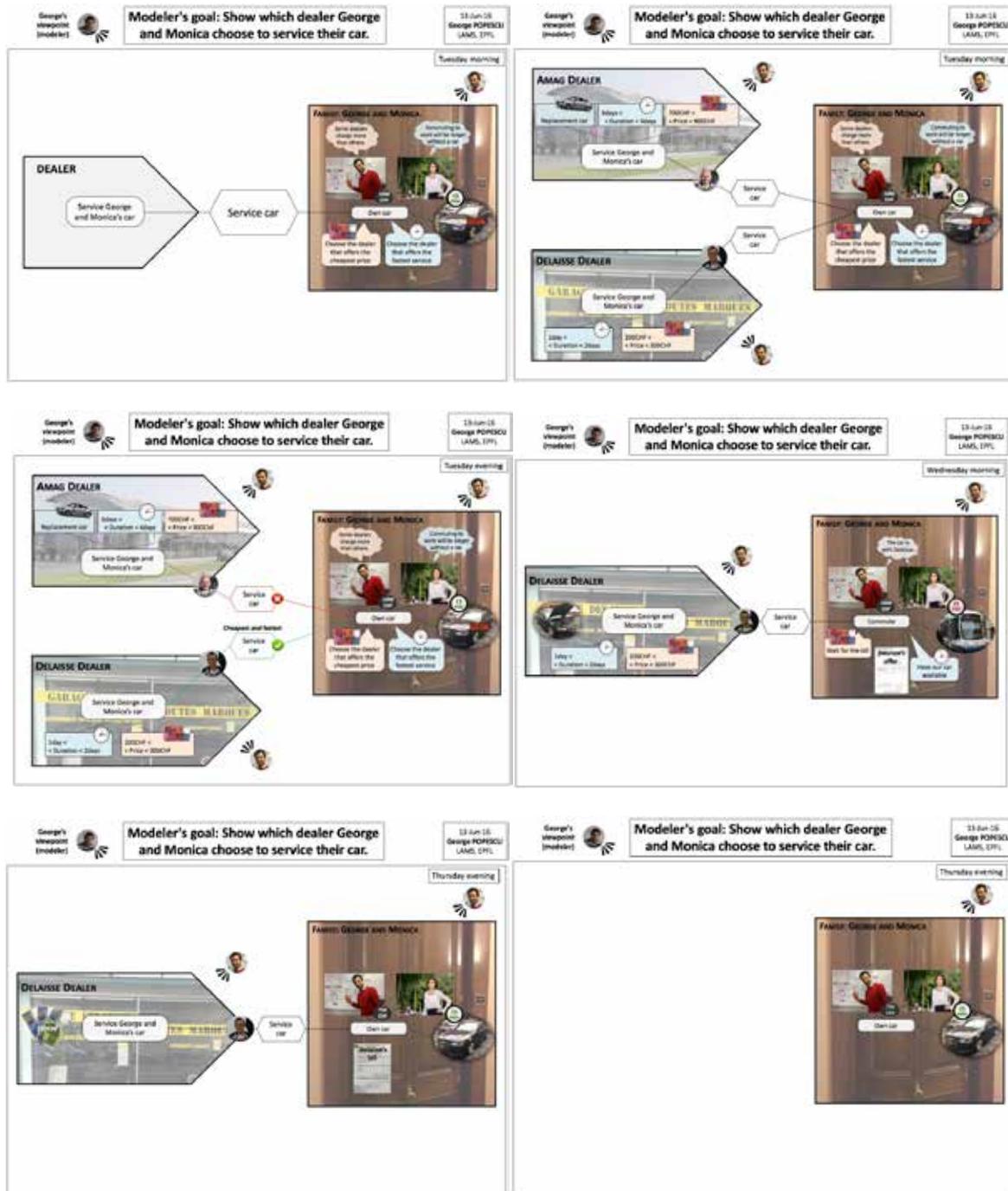


Figure 94: SEAM car-maintenance service model showing viewpoints and visual metaphors

20th iteration

Compared with the proposed model presented in Chapter 4.5.3 the model from Figure 95 includes a criteria evaluation based on pluses and minuses instead of colored smileys. The metaphor for money is the same throughout the model. We also model all car service dealers in Geneva with a cloud to show the relation with the two customers' beliefs (what they are looking for).



Figure 95: SEAM car-maintenance service model showing the Questions, Options and Criteria mapping

Appendix 2 Case of Meeting Scheduler with i^*

We include other model iterations for the case of meeting scheduler modeled with the i^* notation. They are useful to compare against the proposed model iterations from Chapter 5.5.3 in terms of choice of graphical elements and terminology used across model instances.

3rd iteration

Compared with the proposed model presented in Chapter 5.3 the model from Figure 96 uses colors to distinguish between goals, softgoals, tasks, sub-tasks and resources. This notation was proposed by (Moody, Heymans, & Matulevicius, 2009). The complexity of the model increases from one model instance to the next with the gradual addition of graphical elements, as shown with the legend.

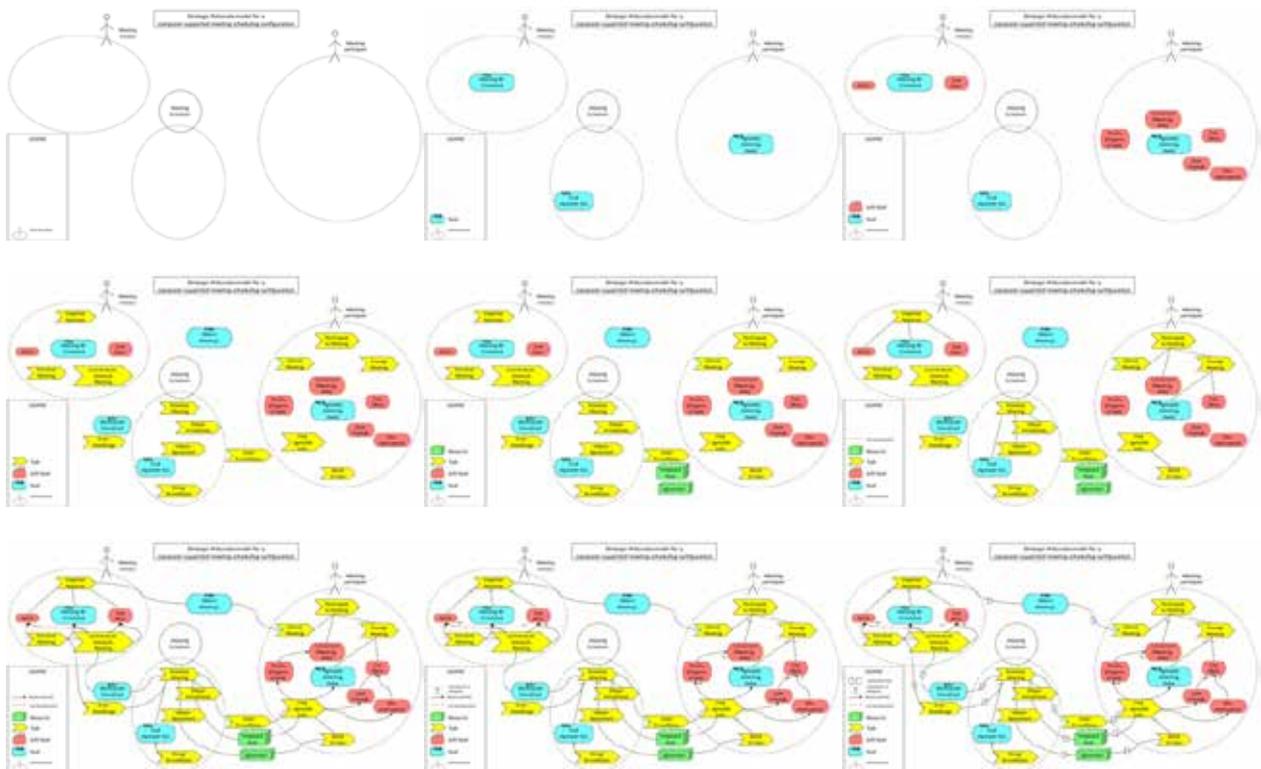


Figure 96: i^* meeting scheduler model using color coding

6th iteration

The model from Figure 97 is not an i* model. It is a SEAM model of the meeting scheduler. This iteration was particularly helpful to identify elements that are specific for the i* notation that cannot be modeled with the SEAM notation and vice-versa. As shown below, the story was more clear to readers, but the modeler needed to redo the model by using the i* notation, not the SEAM notation. We noticed that there was no question shown in the i* model. This was added with a cloud drawn with yellow background and red boundary, placed between the meeting initiator and the participants. This question was later added in the proposed model from Chapter 5.5.3.

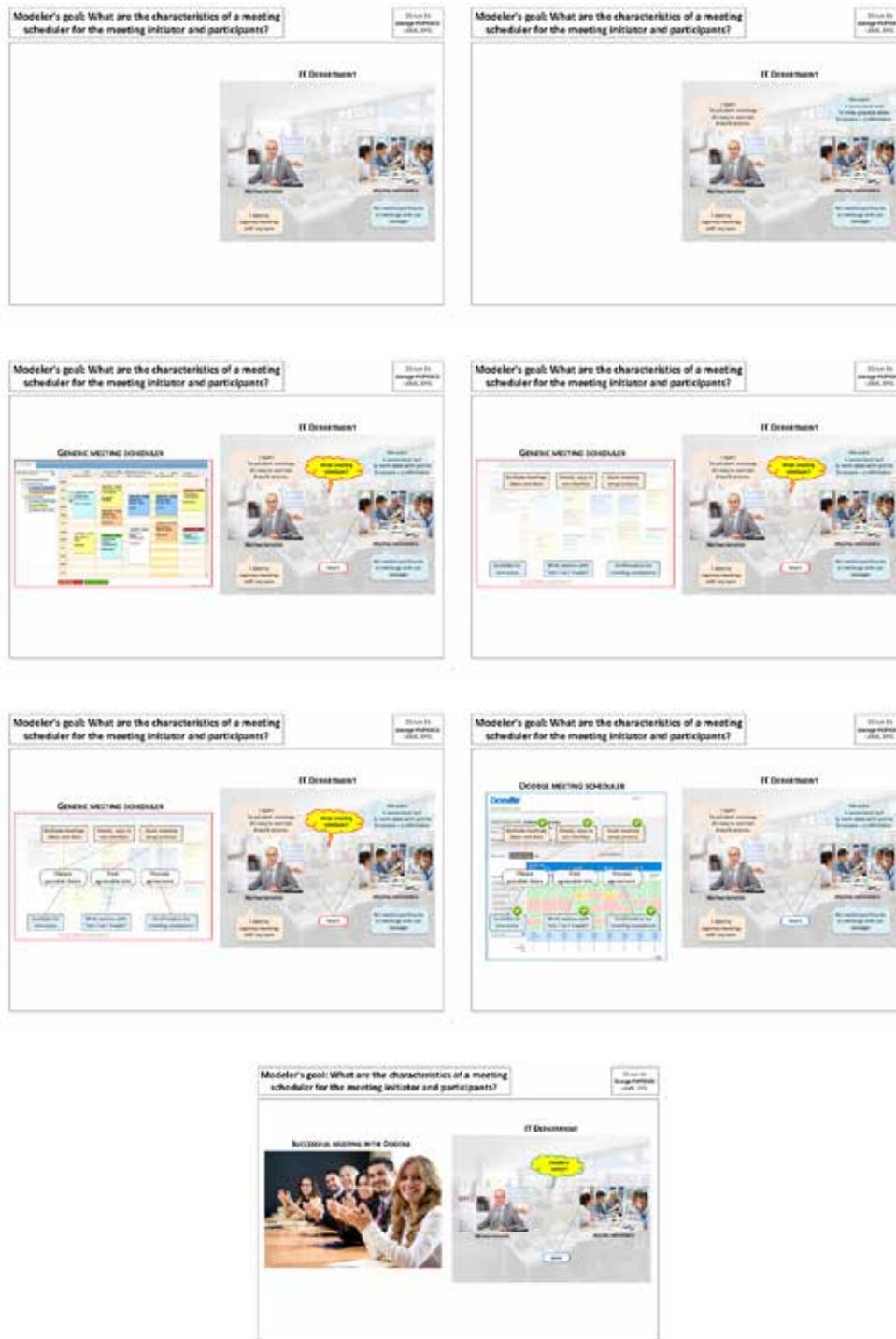


Figure 97: SEAM meeting scheduler model

11th iteration

The model from Figure 99 builds on the initial i* model from Chapter 5.3. All graphical elements are shown in the same structure and with the same terminology. Two possible evaluations of the characteristics of the meeting scheduler are considered: to the left, a binary one, marked with red and green arrows and icons, and to the right, another one, marked with red and green arrows and smileys. This exploration was useful to draw, in the proposed model from Chapter 5.5.3, the characteristics of the meeting scheduler that correspond with the softgoals of the meeting initiator and participants, respectively. Readers appreciated more the notation based on icons (check-marks) rather than the one based on smileys.

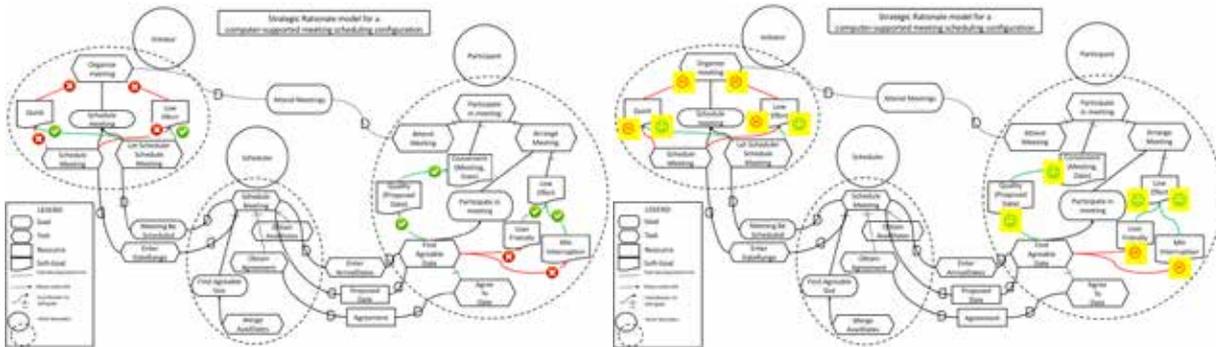


Figure 99: i* meeting scheduler model with add-ins for the rationale – adapted from (Yu E. S., 1997)

Appendix 3 Case of Business Strategy at SITRA with SEAM

The SEAM workshop notation presented in Figure 6 can be mapped to the academic notation presented in Figure 100. Here we include all actors in all three hierarchical levels: the customer world, the tourist office value network, and the SITRA value network. The reader will notice the fact that the SITRA value network is modeled both as a whole (showing provided services) and as a composite (showing component organizations with their provided services). In the customer world, there are two types of customers represented with a family and a journalist. In the tourist office value network, all local organizations, associations and institutions are shown with their provided services to mark their contributions to diffuse the touristic information to the tourists. The information diffusion is split into: developing the business offer, organizing the information and collecting the information. As such, services provided by each partner actor are linked to the respective process. In the SITRA value network shown as a whole services are shown using the ITIL (ITIL, 2016) categorization. In the SITRA value network shown as a composite, the component organizations with their provided services are shown to participate in the implementation of the ITIL services.

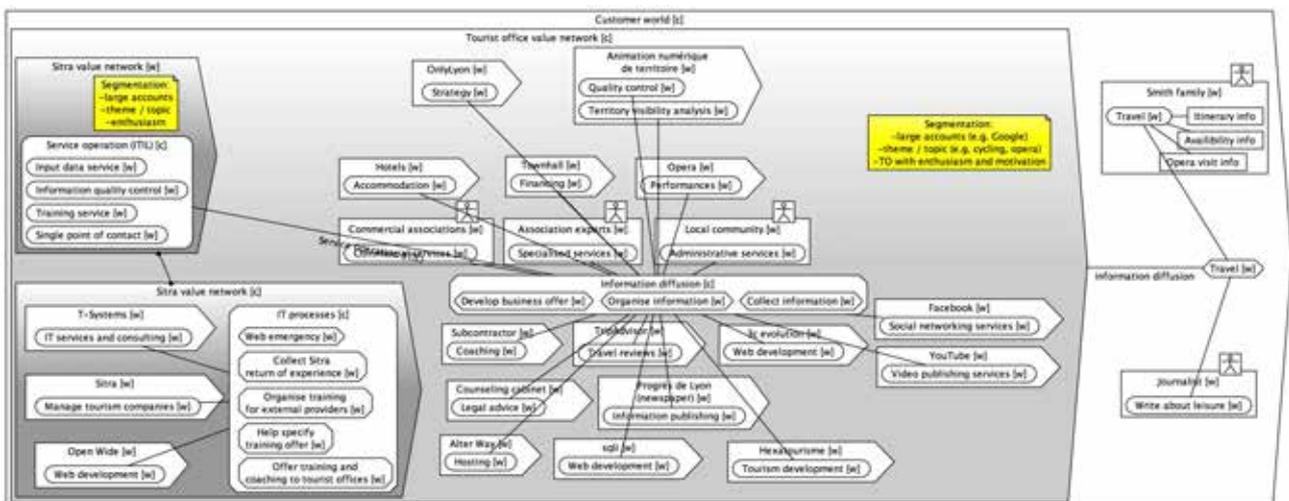


Figure 100: SEAM model of SITRA using the academic notation

An alternative model using the SEAM notation and photos / logos of various was created by the thesis author and is included in Figure 101. System boundaries from the previous model are shown with skewed lines, similar to the workshop notation from Figure 6. The three systems are clearly identified at the top, from left to right: SITRA value network, tourist office value network and the customer world. Color-coding corresponds to each of the three: blue to the left, red in the middle and green to the right. Graphical symbols are shown like in SEAM: ovals for services, hexagons for processes, rectangles for properties, etc. For simplicity, relations between services and processes are not shown as well as the boundaries of each system, e.g. hotels, town hall, association experts, etc. Systems have simplified graphical representations and are shown with a photo / logo and the service they provide, but without their boundaries. The overlay of the services on the boundary shows that the entire organization (positioned to the left) offers these services (e.g. “IT services” / “Information diffusion service”) to the organization positioned to the right.

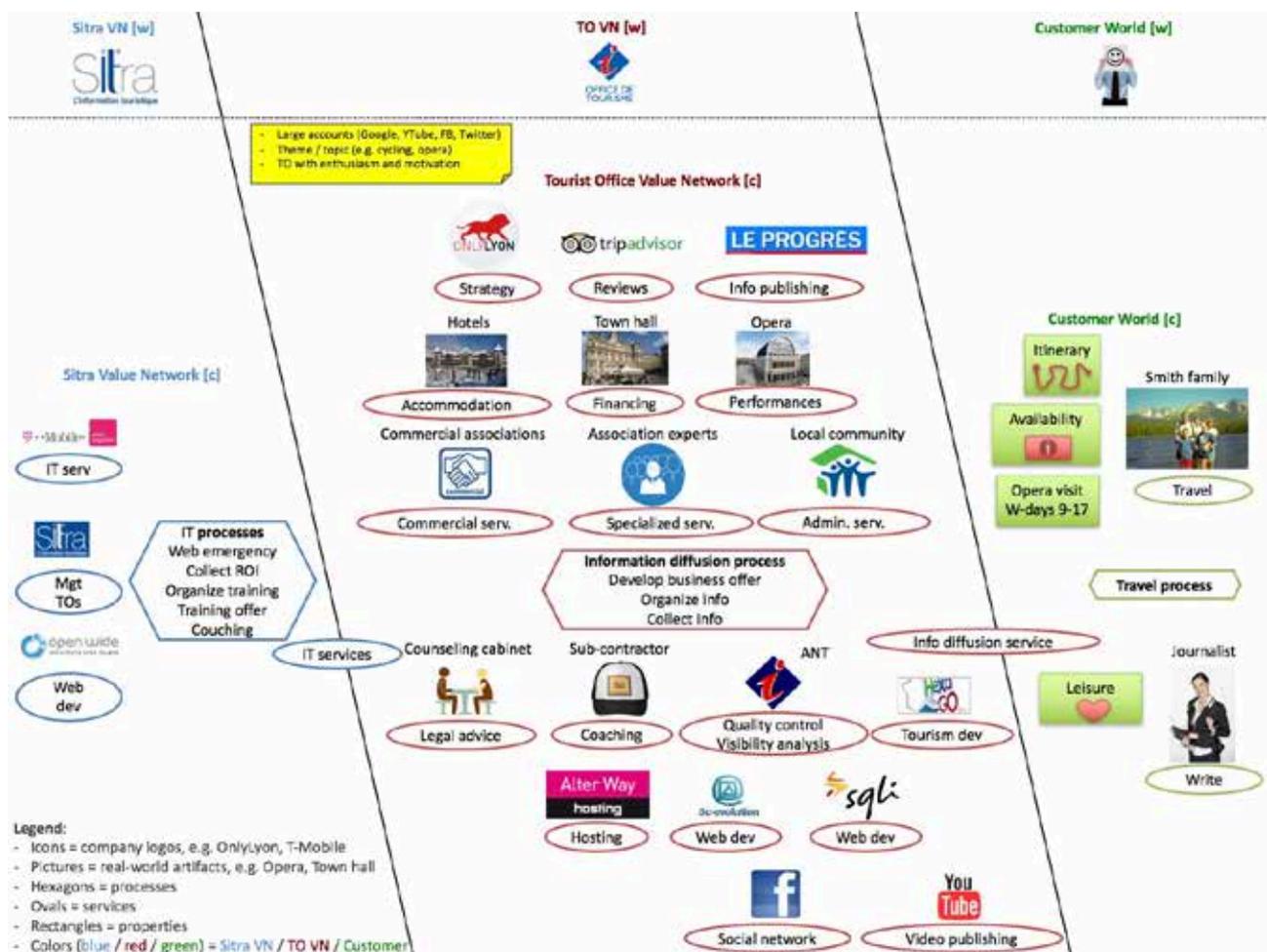


Figure 101: SEAM model of SITRA using a notation based on photos and logos

In another simplified representation of the SITRA case (Figure 102), we maintain the SEAM notation using systems' boundaries. We model the tourist segment using a visual metaphor and show two segments of tourists ("Michael", a mountain cyclist, and "The Smiths", a family of four) together with their goals. The tourist office value network is shown both as whole (providing a service) and as composite (with the details of the process implementing this service). The "SITRA value network" is part of the tourist office value network and provides "IT services". Again, the implementation of these services is shown in the "SITRA value network" shown as a composite.

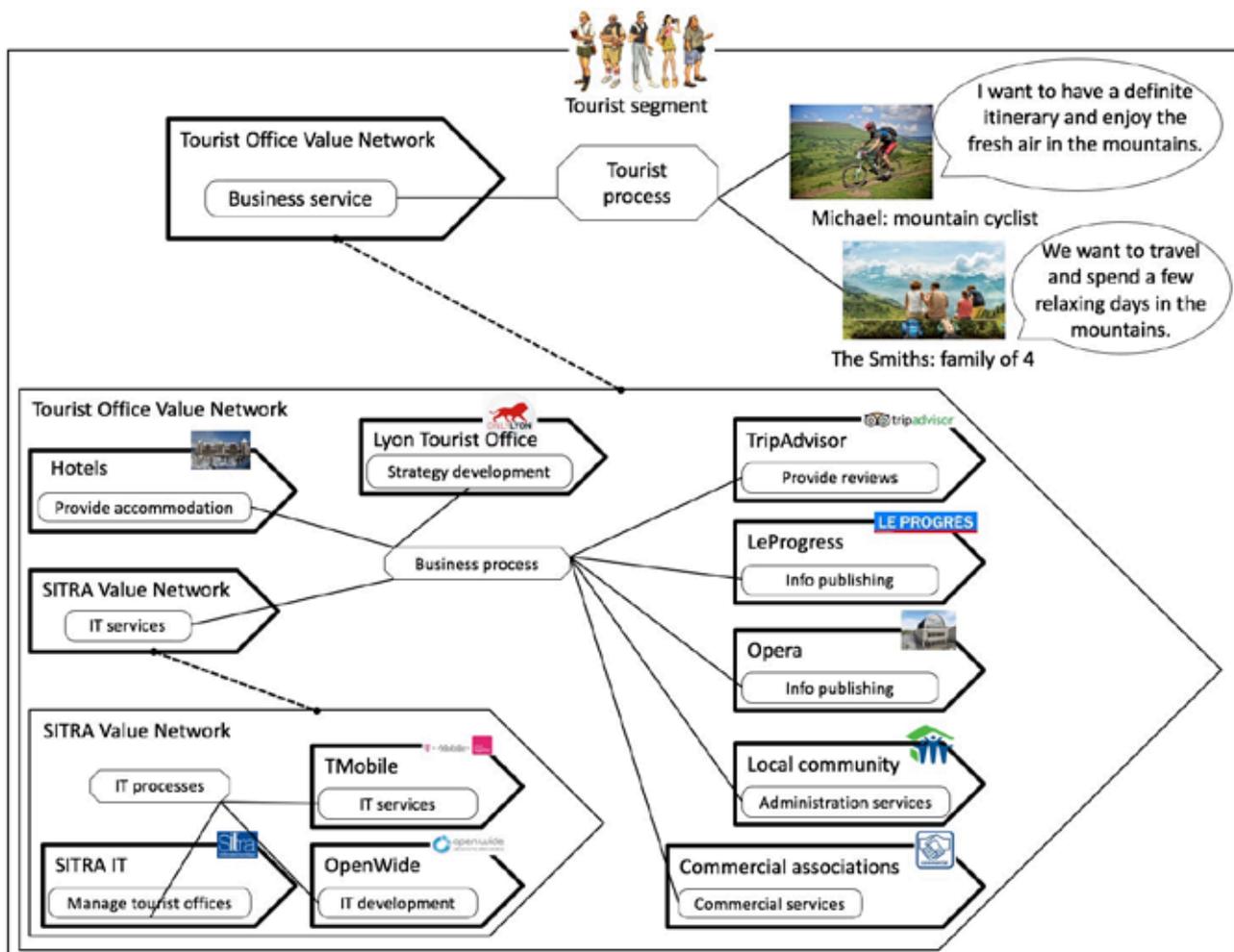


Figure 102: SEAM model of SITRA combining Behaviour and Goal-Belief modeling

Another model was done by a professional designer (courtesy to (Wegmann A. , 2012)). In the Figure 103, all graphical elements and conventions used are explained with text. The key idea is that all tourist offices of SITRA will share the same list of hotels and events, such that the tourists will find coherent information. They need to both compete and cooperate. In the center, tourist offices collaborate together and have different competencies to develop offers and welcome tourists. Towards the right, in order to attract clients, information has to be given to them through various means. The role of SITRA, shown to the left, is to provide services (training, common thinking, guidance) such that the local economy will grow.

There are three systems of interest. First, to the right, the client world shows different targeted groups: groups of tourists arriving by bus, families arriving at a touristic location by family car, individuals dreaming of visiting a castle, practicing skiing or enjoying sunny days in the mountains,

users browsing the internet for information, etc. These clients observe the services provided by the tourist office value network: museum visits, rural tourism, local market events, accommodation, etc. Second, in the middle, the activities proposed by the tourist office value network are highlighted. From top to bottom, they are: snow sports, rural tourist, museum visits, church and monastery visits, hiking in the mountains, fun-parks, hotel accommodation, local markets, and group visits. The members of the tourist office value network observe the operation services provided by the SITRA organization: input data service, information quality control, training, and single point of contact. Third, to the left, the SITRA value network shows the strategic committee group, a city and other IT companies providing products / services for the SITRA organization, which contains IT teams working closely with others (e.g., SEAM workshop facilitator, companies providing storage services, consulting, web development, etc.).

The connections between the systems of interest from above are shown with various types of bridges to symbolize the different creative activities that Sitra proposes to attract clients. Also, a plane distributes information from the value network to the final users in order to attract and convince them to explore the Sitra proposed activities in the middle. There are three centers which propagate information: Sitra IT makes the information available online, tourist offices offer maps and guidance with respect to places worth exploring nearby and Sitra tourist officers offer support to tourists by word of mouth. In addition, information is also passed further on from clients who return from the “Sitra world” to friends, families, neighbors, etc.

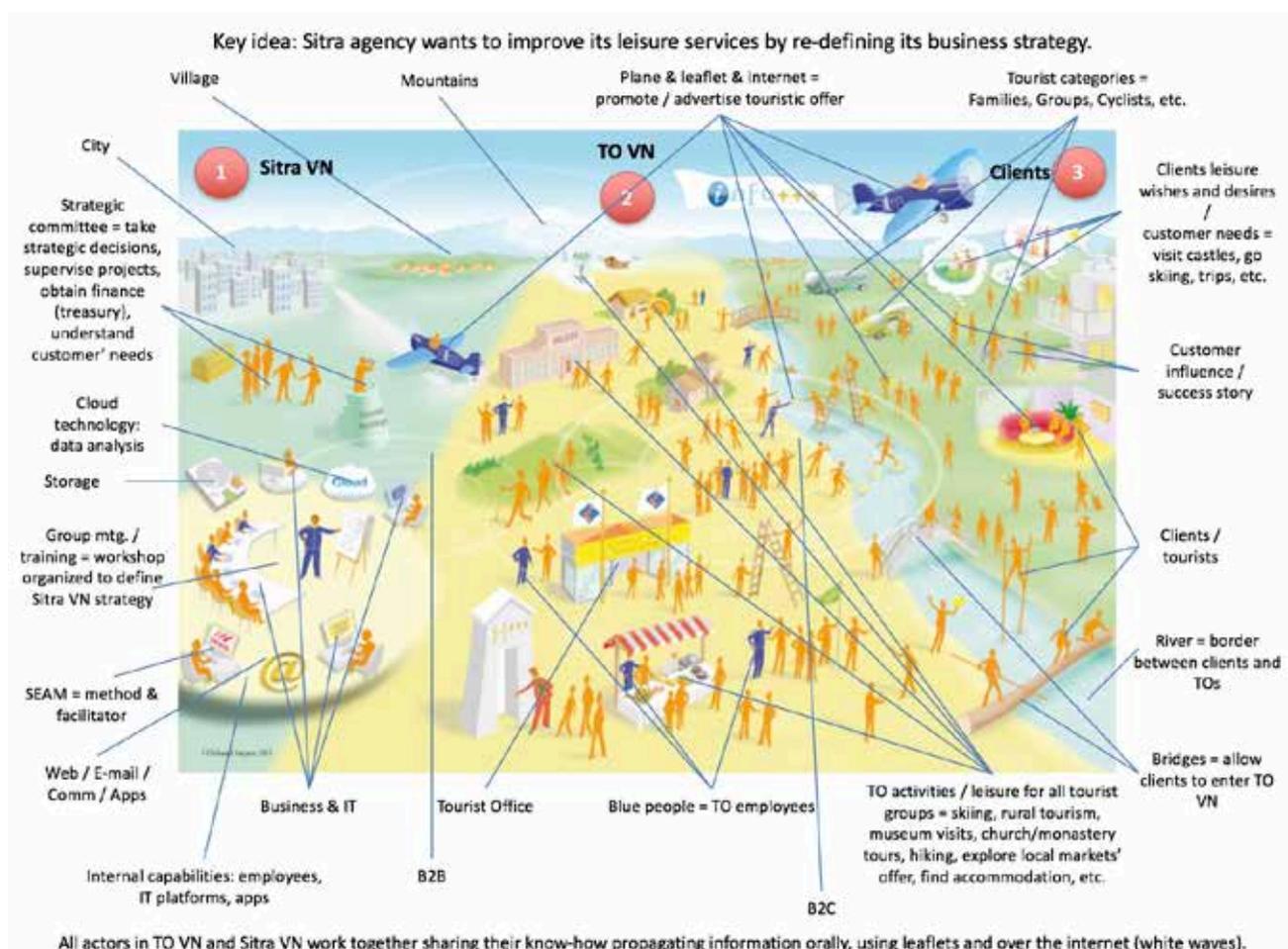


Figure 103: SEAM model of SITRA using the artwork of a professional designer

Professor Alain Wegmann, in his role as SEAM modeler working for SITRA, created the below models to tell the story (original text in French below) of the new organizational strategy developed during two workshops with the members of SITRA, a French touristic organization. Our goal, as designers of modeling notations, is to identify the implicit elements represented in the created models (Figure 104) by comparing model elements with the oral interpretation recorded by the modeler. For future work, we plan to develop improved models that make the story and the content more explicit, based on the recommendations presented in this thesis.

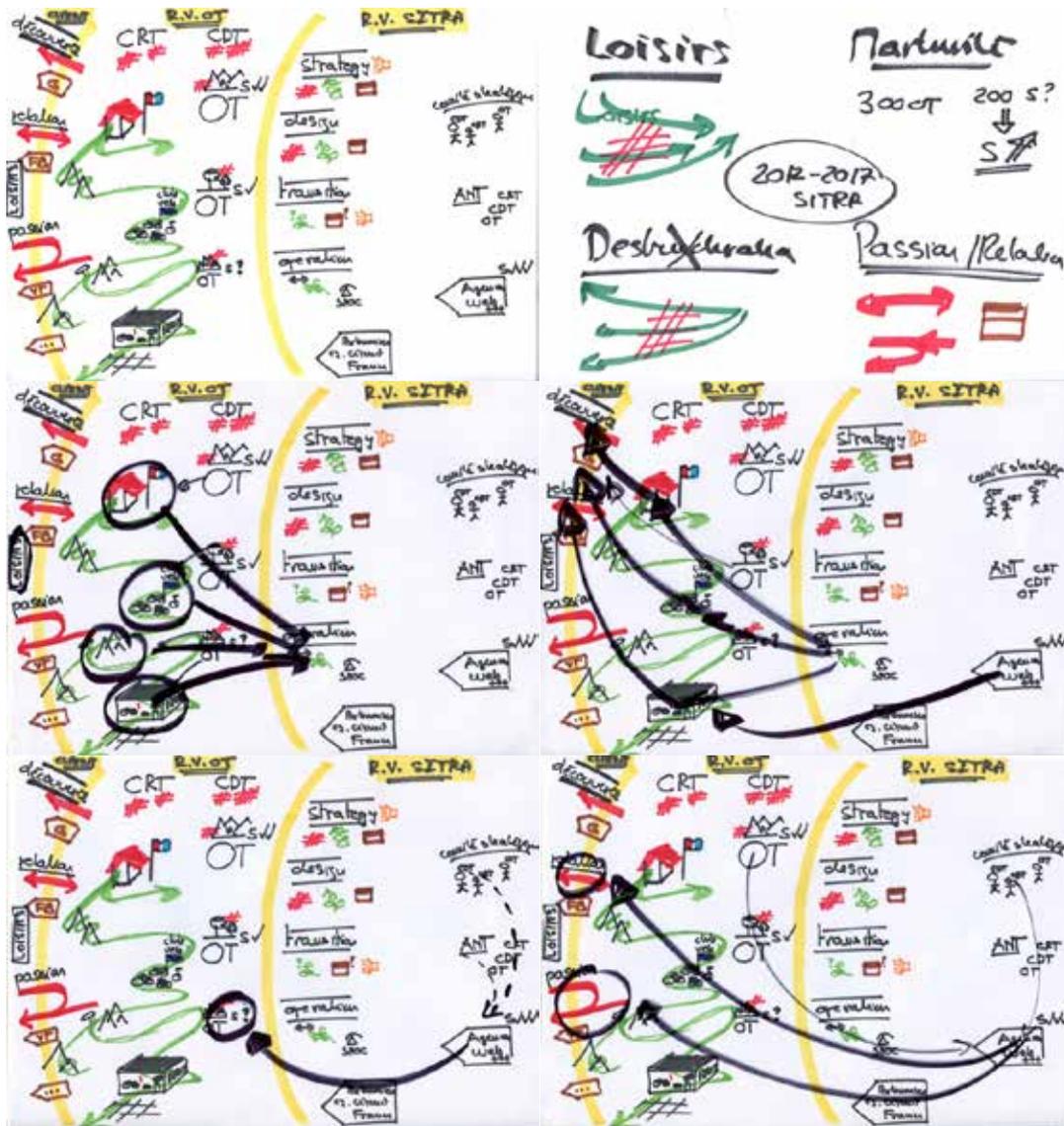


Figure 104: SEAM model of business strategy at SITRA – Context, conflict and climax – source: (Wegmann A. , 2012)

Modeler’s story

“Cette présentation illustre les résultats des ateliers stratégiques de SITRA qui ont eu lieu en mai et juin en 2012 à Chambéry et à Grenoble. Le but était de définir la stratégie de SITRA pour 2012 à 2017. On va utiliser la méthode SEAM qui incite à travailler sur un exemple concret et l’exemple concret est représenté par un modèle que je vais vous montrer. Voyons ici un exemple d’un problème qui est en effet comment on fournit l’information dans le cadre d’une initiative encourageante de vélo, en effet c’est le programme « 1 journée = 1 col » dans lequel on ferme un col par jour et ça permet aux utilisateurs de vélo de se donner à leur sport favori de manière très agréable.

Alors, la première chose qu'on peut dire que nous avons identifié c'est que le monde du touriste évolue et maintenant devient un monde de loisirs puisqu'une partie importante de touristes sont aussi des résidents ou des gens locaux, de la famille, qui viennent plutôt se faire des loisirs donc le tourisme s'éteint. Donc on va parler plus du monde des loisirs.

Dans ce dessin, à gauche ici les clients qui s'occupent des loisirs, qui veulent des loisirs, au milieu ici le monde qui fournit les moyens d'avoir des loisirs et à droite ici, SITRA.

Alors, on voit 3 dimensions importantes :

- découverte : le client cherche à découvrir l'information souvent il le fait par Google
- relation : il développe des relations avec le monde des loisirs et souvent c'est par Facebook
- passion : il développe sa propre information qui illustre sa passion suivant ses vidéos sur YouTube ou éventuellement des sites spécialisés qu'il développe.

Le monde du tourisme maintenant se réorganise pour arriver à adresser le monde des loisirs. Pour faire celle-là il y a des CRT, les CDT et les Offices de Tourisme. Il y a différents types d'office de tourisme : de montagne, de pleine. Ils sont de différentes tailles : soit petites, soit grandes. Ils ont de différents niveaux de maturité par rapport à SITRA. Ces offices de tourisme segmentent leur marché ainsi des CRT et CDT et cette segmentation arrive de structurer l'offre - ici on voit un club de vélo, un marchand de vélo avec son parking devant et une route qui permet de se balader dans ce monde. SITRA qui est à droite ici va offrir des services à ces différents acteurs pour structurer cette offre et pour surtout mettre de l'information dans le système informatique. Et pour faire celle-là on va travailler sur l'aspect stratégique sur le design des services et de l'information, sur la transition et sur l'opération.

Dans ce petit dessin on voit en rouge la segmentation, en vert l'information et en brun les outils technologiques. Les petits hauts parleurs signifient communication. Donc ici on va, au niveau de la stratégie, aider les offices de tourisme à se structurer puisque en effet la structurer de l'information en SITRA permet aux différents offices de tourisme de faire une offre structurée. Donc, quelque part, la structure de l'information aura aussi un rapport avec la structure marketing que les OTs vont mettre en commun. Les OTs sont des entreprises qui ont une certaine compétition puisqu'ils cherchent d'attirer des clients dans leurs territoires mais ils travaillent aussi en coopération puisqu'ils ont intérêt à attirer plus de clients dans leur région que dans les autres régions. Donc SITRA travaille beaucoup sur cette collaboration en mettant en commun ses choses ou ils doivent collaborer et en leurs laissant complète liberté de faire de la concurrence.

Donc au niveau stratégique on cherche ce que doit être fait. Au niveau design on définit et on construit l'offre soit de la technologie soit structurer de l'information soit la segmentation. Transition on est des organisations à migrer, donc surtout en faisant de l'information et en faisant des FAQs. Et, au niveau de l'opération on va en effet fournir le service par faire entrer / sortir de l'information avec la hotline.

A l'intérieur de SITRA on a un comité stratégique, on a des ANT qui sont soit au niveau régional, soit au niveau départemental soit des offices de tourisme et on travaille avec des agences web extrêmement compétentes et alliés entraînent importants ou avec des partenaires qui ont développé des offres.

A présent il y a 4 stratégies qu'on a identifiées. La première stratégie est de dire qu'on va s'adresser au monde du loisir. Et le monde du loisir ça veut dire que c'est tous ces acteurs-là qui vont saisir de l'information : les gens de la campagne, des montagnes de villes, etc. Et donc SITRA va collecter de l'information de cet ensemble d'acteurs et non seulement des OTs. Et donc ça c'est nouveau et c'est très important. Donc les OTs vont devenir importants pour motiver, identifier comment encourager ces différents acteurs à générer de l'information. Une fois que cette information est générée la deuxième stratégie va être de lutter contre la déstructuration - la déstructuration fait que Google devient le moyen privilégié d'obtenir de l'information, donc l'information est un peu partout mais elle est surtout obtenue via Google. Et l'idée est qu'en effet tous ces acteurs ici vont fournir l'information, donc c'est le même principe que la première version de SITRA et maintenant c'est bien au-delà des OTs, bien sûr les OTs vont rester et fournir l'information mais aussi leurs partenaires et donc tous ces partenaires vont obtenir de l'information des OTs. Et pour faire celle-là il va falloir que les agences web et que les partenaires aident à promouvoir ou fournir de la technologie qui permettent de réaliser ce genre de choses et à faire cette transition. Donc ça c'est la deuxième stratégie qui est en effet la stratégie de lutte contre la déstructuration en fournissant de l'information à tous les points possibles au niveau de loisirs. La troisième stratégie c'est une stratégie de maturité puisque certains OTs sont moins matures que d'autres dans l'utilisation de SITRA et donc de nouveau les agences web, sous patronage des comités stratégiques et des ANTs, vont aller aider des OTs à développer cette stratégie à mieux utiliser l'informatique pour faire ce genre de choses. Et la dernière stratégie qui est plus une stratégie à long terme c'est que la croissance à long terme va venir par les relations et par la passion et donc encore une fois les agences web toujours sur le pilotage du comité et en prenant des initiatives d'OTs qui sont plus mur ou qui utilisent l'informatique de manière plus avancée va développer une offre qui permettra de construire les relations avec le client et de permettre au client de développer sa passion.

Donc pour résumer, quatre stratégies ont été développées - un joli résumé ici en quelques mots:

- la première est d'aller au niveau de loisirs et pour aller au niveau de loisirs l'idée c'est qu'on va collecter de l'information du monde du loisir donc on va vraiment élargir le périmètre de SITRA au monde du loisir.
- la deuxième stratégie c'est de lutte contre la déstructuration et là l'idée c'est de dire que de nouveau, en effet c'est la complémentaire de la stratégie précédente, l'information va sortir par tous les acteurs et que ces acteurs vont être analysés et compris grâce à une segmentation que les offices de tourisme vont partager. Et ça c'est en effet l'évolution du business actuel de SITRA.
- ensuite, pour permettre avoir une meilleure couverture c'est l'aspect maturité donc l'idée c'est qu'il y a environ 300 offices de tourisme et sur ces 300 offices de tourisme environ 200 ont un niveau de maturité plutôt bas à l'heure actuelle et l'idée c'est vraiment de faire augmenter le niveau de maturité SITRA sur un nombre important de ces offices de tourisme.
- et le dernier c'est passion et relation c'est vraiment de travailler avec les agences web et avec les amateurs technologiques pour développer le savoir-faire et les outils pour développer cette vision de co-construction de l'information au moyen de technologies web qu'on va représenter par une petite fenêtre Windows, ou plutôt Internet, avec un widget, voilà.

Nous avons ici les 4 stratégies clés :

- focaliser sur les loisirs en prenant tous les acteurs du monde de loisirs

- lutter contre la déstructuration en étant présent sur chacun d'eux
- augmenter la maturité des différents acteurs et
- développer passion et relation avec la technologie

Ça c'est le plan 2012-2017 pour SITRA. Je vous remercie de votre attention !”

Implicit elements

By comparing the story, presented in the above text, against the model elements we identify the following implicit elements:

1. The problem (“Comment on fournit l’information dans le cadre de l’initiative encourageante de vélo: “1 journée = 1 col”?”) together with its details is missing from the model. The details are:
 - a. On ferme un col par jour
 - b. Les utilisateurs de vélo vont trouver très agréable
 - c. Le monde du tourisme devient un monde de loisirs (pour les résidents)
2. Several acronyms are not explained:
 - a. R.V. OT / R.V. SITRA means “réseau valeur” (value network)
 - b. OT means “office de tourisme” (tourist office)
 - c. CRT / CDT mean regional / departmental offices
 - d. G, FB, YT mean Google, FaceBook, YouTube
3. Some relations are explained orally but are not represented graphically:
 - a. The discovery of information (via Google)
 - b. The relation with the hobbies world (via Facebook)
 - c. The passion for following personalized videos (via YouTube and other websites)
4. The re-organization of the tourist world is not shown (it needs to compare the actual state with the future state)
5. The tourist offices are both competing among themselves to attract more clients from the same region and are cooperating to attract more clients from other regions. They use their resources in common. None of these are not shown in the model.
6. In the transition strategy, the modeler mentions the migration but it is not clear of what and how it will be done.
7. The strategy is an extra layer of the presentation not shown graphically but explained throughout. The 4 strategies are all mentioned orally but shown using only black arrows that have no meaning.
 - a. Hobbies: collect the information provided by both the tourist offices and the clients
 - b. Unstructuring: all actors provide information to all clients via Google, Facebook, YouTube and partners
 - c. Maturity: web agencies help the tourist offices to better use the IT for advertising
 - d. Relations and passion: develop client relations using technology for long-term growth

A legend is included but the goal should be to make it unnecessary. It includes the following conventions: # red = segmentation; # green = information; # brown = technology tools; # loudspeakers = communication.

Appendix 4 Case of IT Strategy at EPFL with SEAM

We present the impact of this thesis with a practical example from real life, in which this research was applied. The recommendations for modelers from this research were applied by professor Alain Wegmann, the creator of the SEAM notation, to perform business story-telling (communicate business strategy) and create models for the Vice-Presidency of Information Systems (VPSI) at Ecole Polytechnique Fédérale de Lausanne (EPFL).

EPFL has an IT department of approximately 150 people, called the Vice-Presidency of Information Systems (VPSI). Overall EPFL has 300 people working in IT. The management of VPSI decided to initiate a reorganization project, in order to become more service-oriented. The goal was to infer a strategy, by bringing more value to IT users while having more efficient IT architecture.

SEAM was used as a template for this new organization. It served to write the IT strategy and governance document. The sketch from Figure 105 was used to capture management dimensions necessary to run the VPSI service organization, but was not shown to stakeholders. It is useful only for the modeler as a graphical metaphor summarizing SEAM principles.

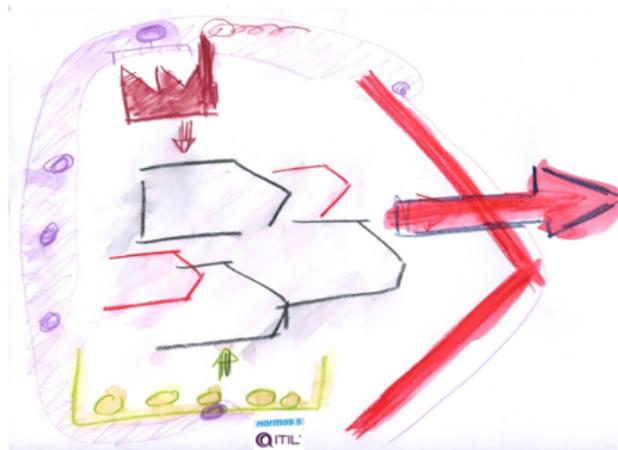


Figure 105: SEAM service organization template sketch

The modeler, Professor Alain Wegmann, used the example of an IT project to show the story of the IT strategy at EPFL. The example considered the communication of the VPSI strategy to all IT stakeholders. It focused on the understanding of the computer-classrooms' infrastructure. The goal of the modeler was to enable his readers (mostly system administrators) to understand the future IT strategy of EPFL using the example of the Virtual Desktop Infrastructure (VDI) IT project.

He wanted to show the following story: A federated infrastructure that can manage thousands of virtual machines needs to be designed in two steps. First, a technical service, named VDI, is developed and offered by the Service Factory to the computer classrooms administrator. It replaces the infrastructure managed in each school and removes the risks and costs from the classroom administrators. Second, a business service, named computer classroom, is developed and offered by the schools' IT departments to teachers and students. The teachers do not need to get involved in the implementation of the service, they benefit from the value of the service. Other projects, besides VDI, can benefit from the same IT and business organizations' definition-pattern by showing what IT services bring to users (adapted from the slides of Prof. Alain Wegmann).

The storyboard model from Figure 106 was shown at the FORUM SI (Systèmes d'Information), the annual meeting of VPSI collaborators. It was evaluated by the vice-president of information systems of EPFL, professor Karl Aberer, the vice-president delegate, Mr. Didier Rey, and several system administrators. The Vice-President Delegate of VPSI, Mr. Didier Rey, mentioned that "The SEAM approach was essential to understand how to articulate all the management dimensions found in a service organization. It allowed us to make it concrete and understandable."

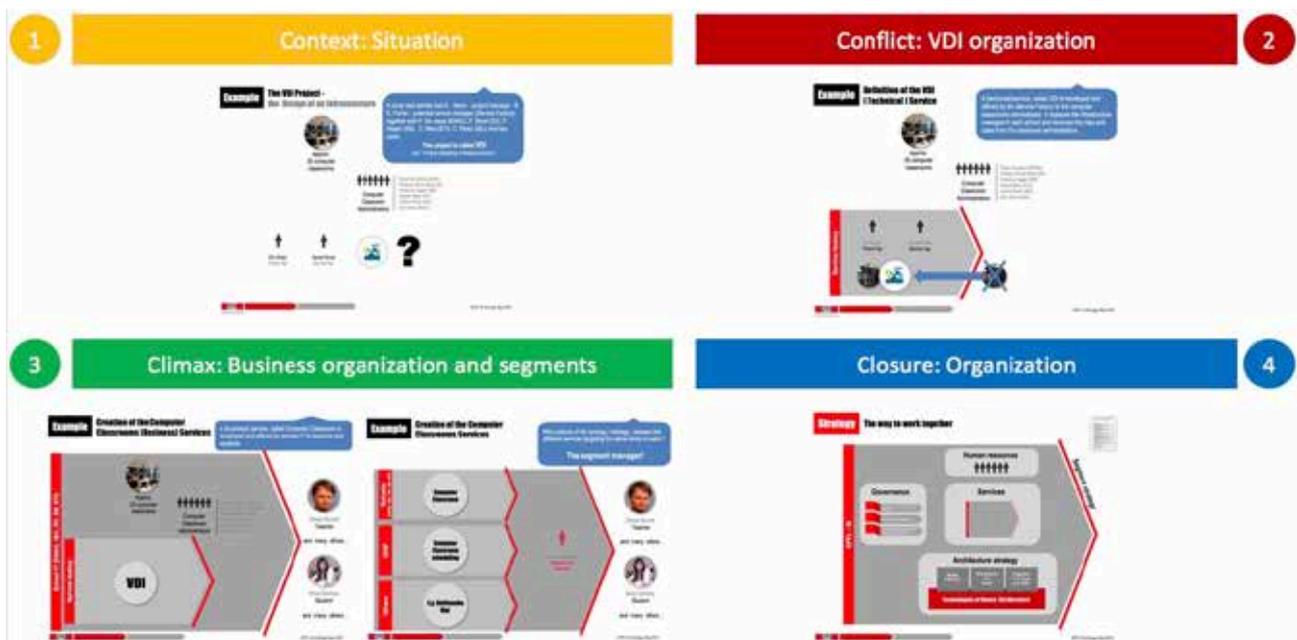


Figure 106: Storyboard of SEAM model instances showing the IT strategy at EPFL

The strategy shows the intention of organizational stakeholders for the projects they manage. It helps explain the reasoning leading to results (choice). Rationale can be used to offer more context about the choice, for instance describe the environment in which the strategy is defined. The definition of strategy is similar to the modeling of choice, as shown for SEAM. It enables workshop stakeholders to reflect about alternatives. Strategy is thought of as the necessary allocation of resources to solve a certain problem. It is related not only to the choice of an alternative but also to the choice for not pursuing the other alternative(s). For EPFL, the IT strategy focused on choosing an operational mode based on virtual machines. The alternative that was not chosen was based on physical machines. The workshop participants' discussion about the observed reality demands for concreteness. This concreteness can be shown in the model by using graphical elements that match the observed reality.

Context

The context is shown with one model instance (Figure 107). It includes the actors and technology that are needed to work on the VDI project. The people involved in an initial study of VDI are mentioned in the blue call-out.

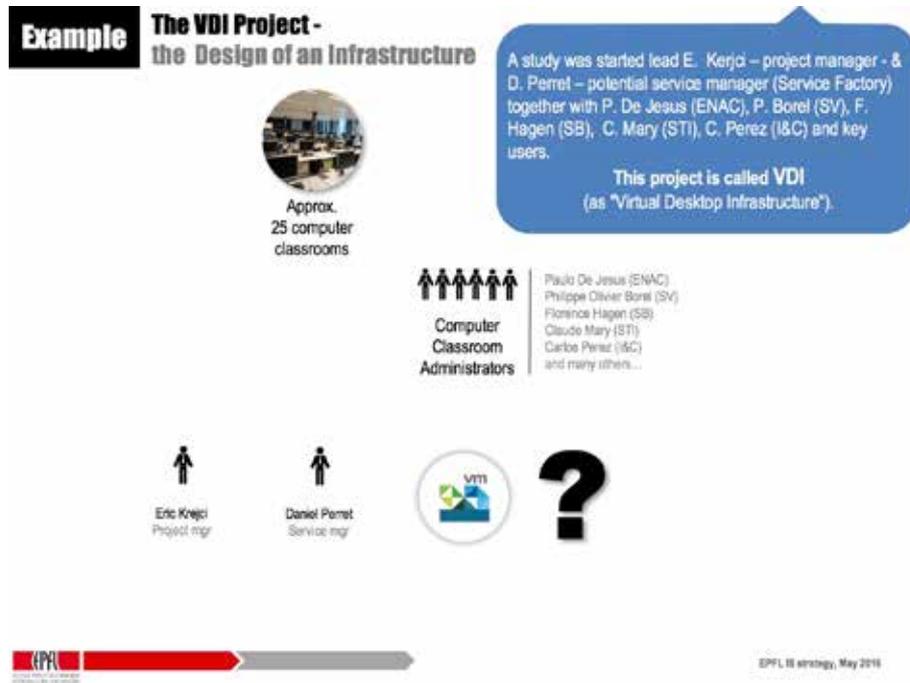


Figure 107: SEAM model of IT strategy at EPFL - Context – source: (Wegmann & Rey, 2016)

Conflict

The conflict is shown with one model instance (Figure 108) as well. It shows how the technical service (virtualization, or VDI) is developed by an organization named “Service Factory” to replace the physical machines. This service is offered to computer classroom administrators positioned outside the boundary. The advantage is that it removes the risks and costs for administrators.

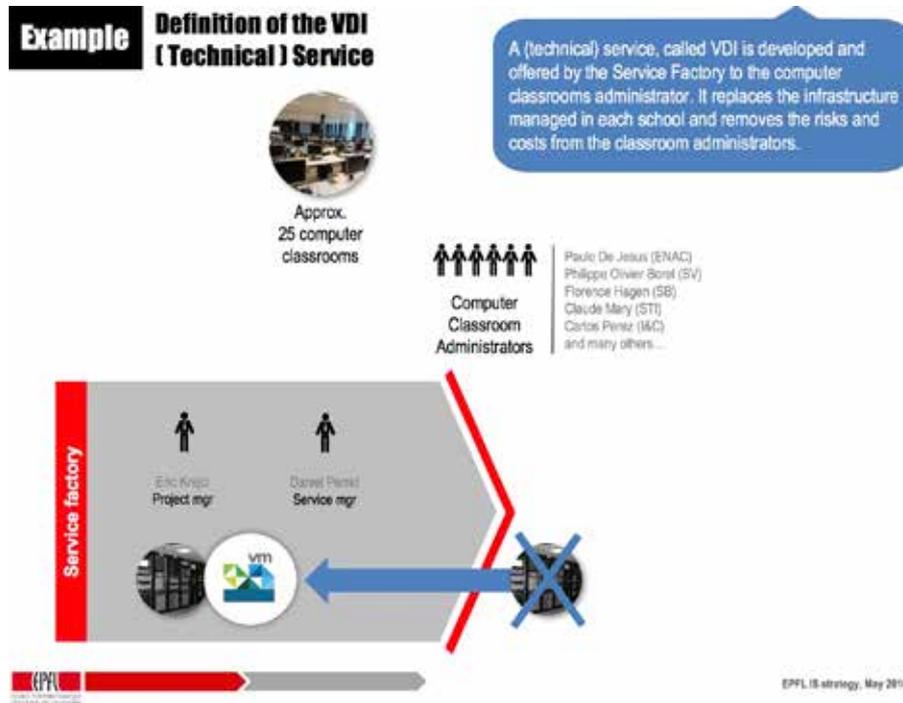


Figure 108: SEAM model of IT strategy at EPFL - Conflict – source: (Wegmann & Rey, 2016)

Climax

The climax is shown with two model instances (Figure 109 and Figure 110). In the first one, a business service (“computer classroom”) is offered by the school’s (e.g., ENAC, I&C, SB, STI) IT departments to two segments of users: teachers (represented by Olivier) and students (represented by Nina). In the second one, a segment manager is in charge of different other related services (“scheduling” and “multimedia management”) targeting the same types of users.

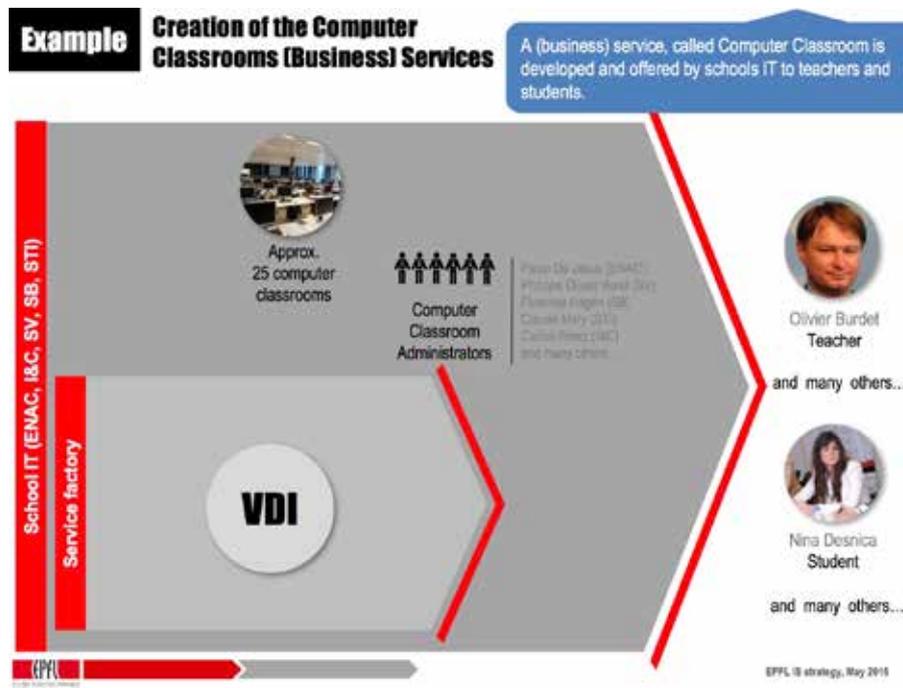


Figure 109: SEAM model of IT Strategy at EPFL - Climax A – source: (Wegmann & Rey, 2016)

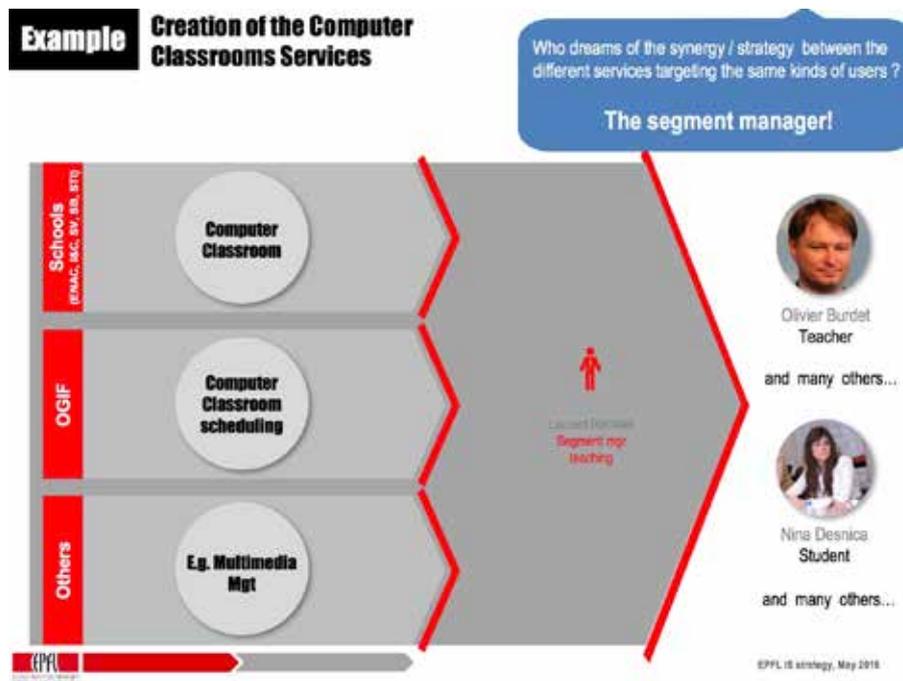


Figure 110: SEAM model of IT Strategy at EPFL - Climax B – source: (Wegmann & Rey, 2016)

Closure

Finally, the closure is shown with one model instance (Figure 111). It emphasizes the strategy expressed with the way collaborators should work together. It shows the segment strategy structure of the VPSI projects that includes the governance or management, the human resources, the architecture or technical strategy and the services offered to users.

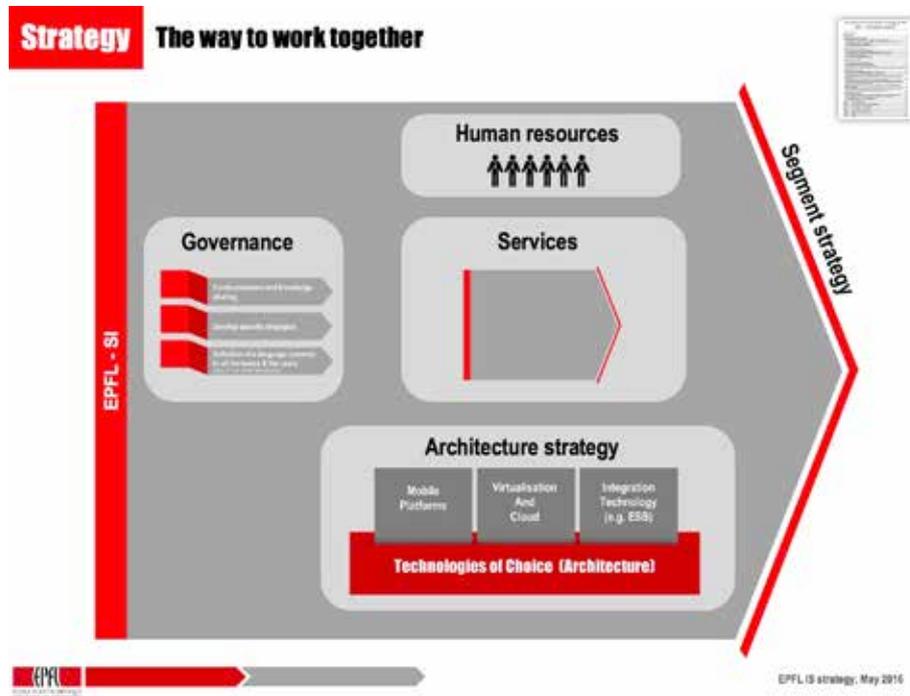


Figure 111: SEAM model of IT Strategy at EPFL - Closure – source: (Wegmann & Rey, 2016)

Another model was done by the thesis author as an alternative to the above, in which SEAM's notation identity is better preserved (e.g., by showing services, processes, goals). The model contains three model instances and follows the SEAM story-telling style: a technical organization provides a technical service to a business organization that provides a business service to two customer segments. These instances are shown in Figure 112, Figure 113, and Figure 114.

Context and conflict

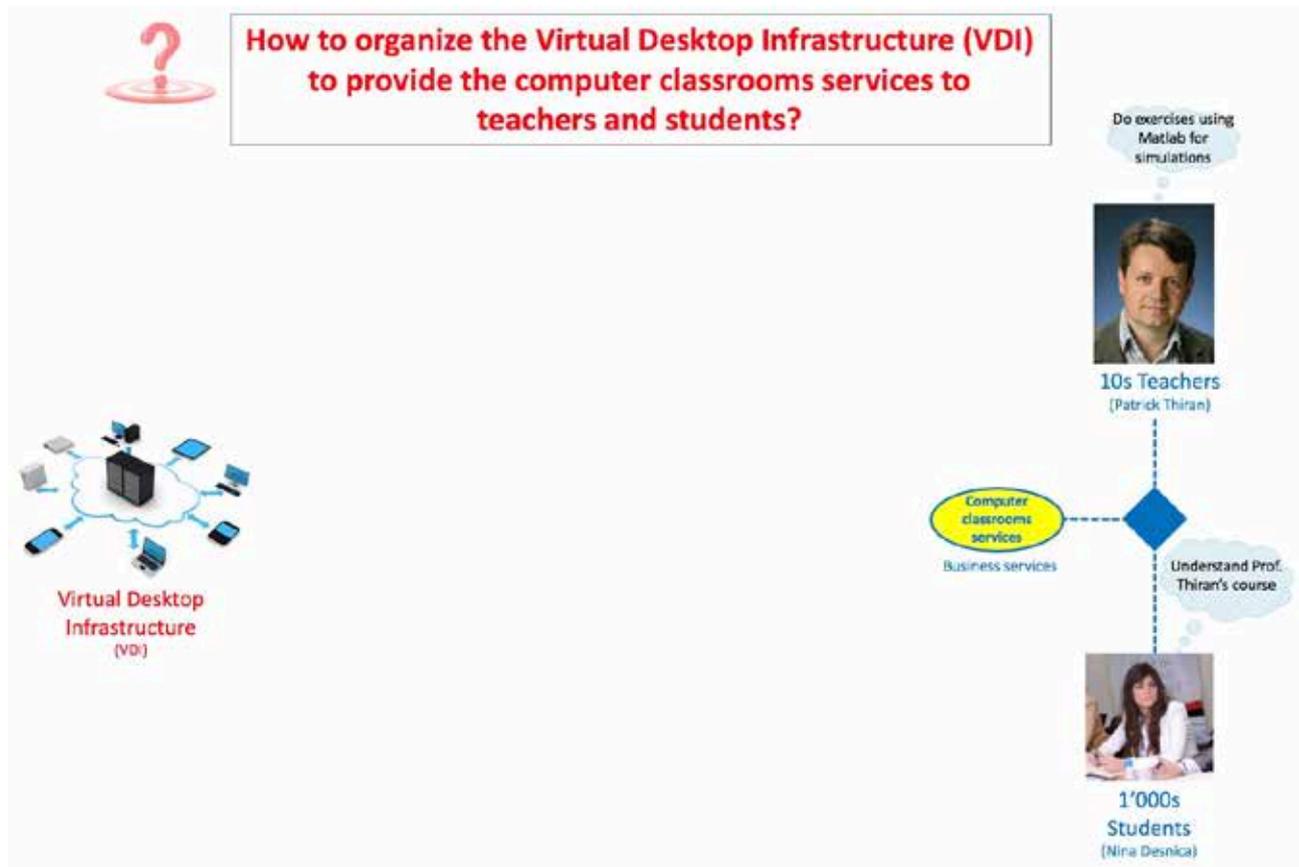


Figure 112: SEAM model of IT Strategy at EPFL – Problem

Climax and closure

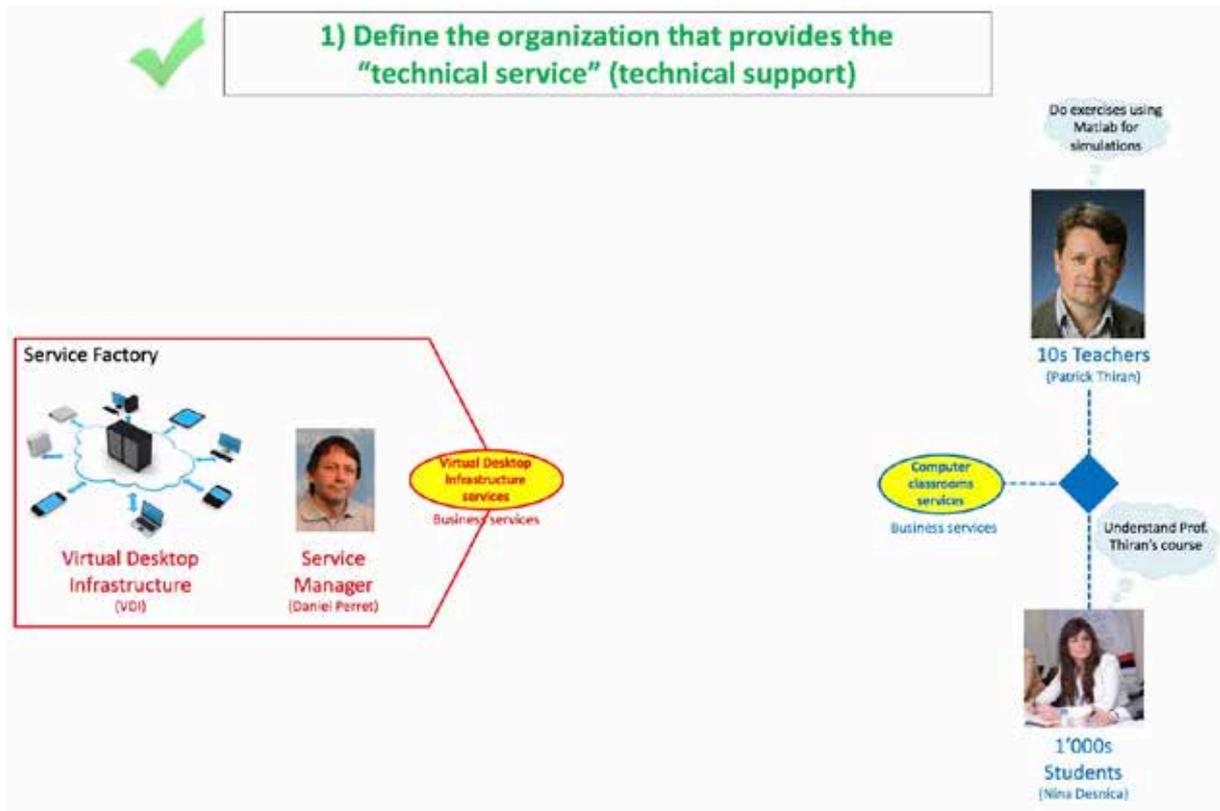


Figure 113: SEAM model of IT Strategy at EPFL – First step of the solution

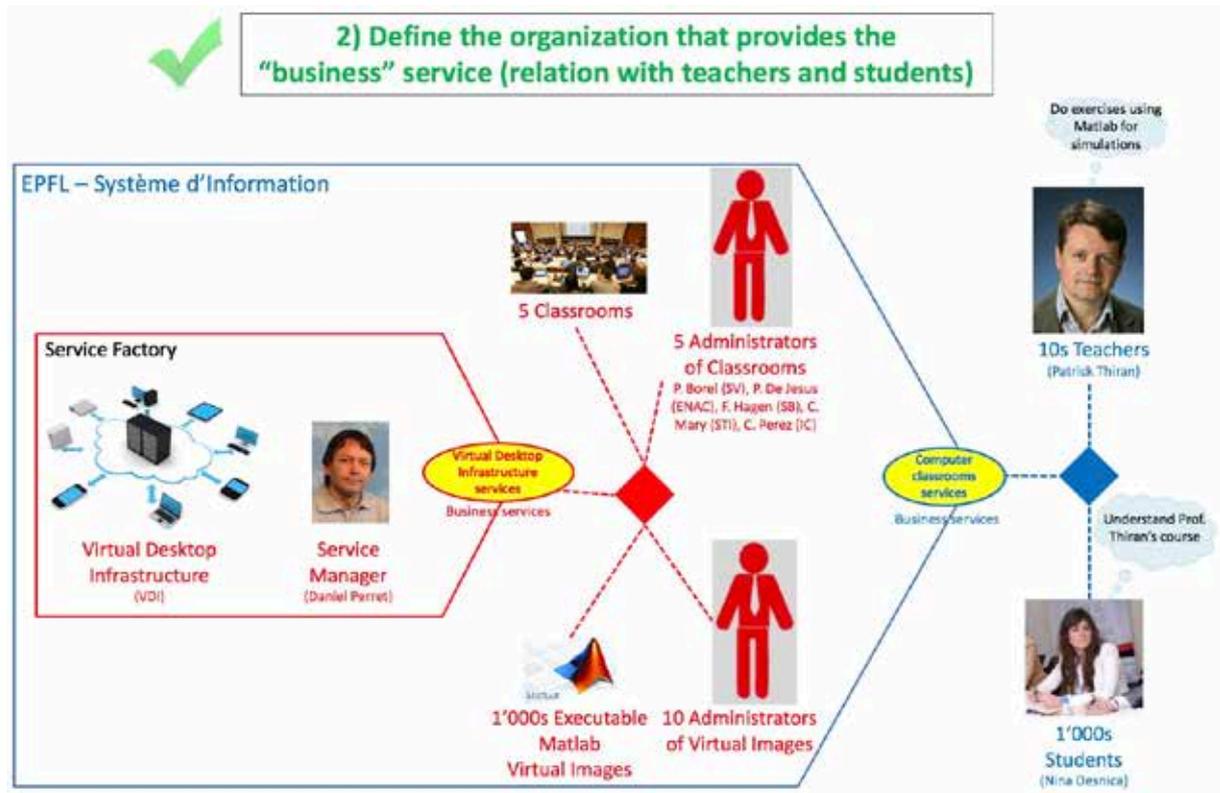


Figure 114: SEAM model of IT Strategy at EPFL – Second step of the solution

Appendix 5 Current and proposed SEAM Evaluation with the Physics of Notations

We evaluate the current SEAM notation with the state-of-the-art notation-evaluation framework, Physics of Notations (PoN). The SEAM notation scores high for some PoN principles: complexity management, graphic economy, and cognitive fit (Popescu & Wegmann, Using The Physics of Notations Theory to Evaluate the Visual Notation of SEAM, 2014). For other PoN principles, such as visual expressiveness, transparency, and discriminability, SEAM scores low. A detailed analysis with explanations is included in Table 12.

Score	Guideline	Definition	Explanation
6	Semiotic clarity	There should be a 1:1 correspondence between semantic constructs and graphical symbols.	Trade-off between symbols' redundancy, overload, excess and deficit for all three types of SEAM models.
8	Perceptual discriminability	Different symbols should be clearly distinguishable from each other.	Based on shape, text, color (user defined), but not based on size, unique values and labels.
6	Semantic transparency	Use graphical symbols that suggest their meaning.	Limited perceptual resemblance and few common logical properties, metaphors and learned associations.
10	Complexity management	Include explicit mechanisms for dealing with complexity.	Value-network used for grouping, functional and organizational hierarchies.
9	Cognitive integration	Include explicit mechanisms to integrate information from different diagrams.	Directly related elements from sub-models (wholes and composites).
6	Visual expressiveness	Use the full range and capacity of visual variables.	Position, color, shape and text but not size, brightness, texture and orientation.
10	Dual coding	Use text to complement (not replace) graphics.	Short labels and description sentences.
8	Graphic economy	The number of different graphical symbols should be cognitively manageable.	Moderate graphic complexity.
5	Cognitive fit	Use different visual dialects for different tasks and audiences.	No expert / novice user discrimination.
63	Total		Improvements: use of color, metaphoric icons, expert / user discrimination.

Table 12: Current SEAM notation evaluation with PoN

For instance, the first principle (semiotic clarity) states that there should be a 1-to-1 correspondence between semantic constructs (entities) and graphical symbols. However, models could include different graphical symbols for the same entities or the same graphical symbols for different entities. This creates confusion and ambiguity. Semiotic clarity implies that if entities are at different hierarchical levels then they need to be represented by a different graphical symbol. Therefore, a different notation is needed for each level. SEAM semantic constructs are hierarchical (wholes inside com-

posites), whereas entities are generic and represented with several types of graphical symbols (Chapter 2.1.1).

We present below the evaluation with the Physics of Notations (PoN) of the proposed SEAM in contrast with that of the current SEAM. The proposed SEAM scores higher for all principles except for graphic economy. Indeed, our proposed model is more understandable by readers but at the cost of using more graphical symbols.

Score	Guideline	Definition	Explanation
10	Semiotic clarity	There should be a 1:1 correspondence between semantic constructs and graphical symbols.	No symbol redundancy, no deficit, overload or excess.
9	Perceptual discriminability	Different symbols should be clearly distinguishable from each other.	Based on shape, color, unique values, labels but not based on size.
10	Semantic transparency	Use graphical symbols that suggest their meaning.	Perceptual resemblance, common logical properties, metaphors and learned associations.
10	Complexity management	Include explicit mechanisms for dealing with complexity.	Grouping each supplier, family – show only one level
10	Cognitive integration	Include explicit mechanisms to integrate information from different diagrams.	Include directly related elements from the Delaisse dealer seen as wholes and composite.
9	Visual expressiveness	Use the full range and capacity of visual variables.	Use position, color, brightness, shape, texture, orientation and text style but not size.
10	Dual coding	Use text to complement (not replace) graphics.	Short labels for all icons.
4	Graphic economy	The number of different graphical symbols should be cognitively manageable.	Moderate graphic complexity.
5	Cognitive fit	Use different visual dialects for different tasks and audiences.	Experts create models for novices.
77	Total		Improvements: use of size, lesser icons, simpler model.

Table 13: Proposed SEAM notation evaluation with PoN

PoN helps evaluate only model instances, one at a time, and recommends improvements for making graphical elements more “cognitively effective”. However, it does not help the modeler design an understandable story. PoN lacks analysis capabilities regarding the entire model, composed of a sequence of model instances that show a story. We consider that it is this story that contributes to readers’ understanding of a model, much more than the design of individual graphical artifacts. A model is the support for the modeler to convey a story, much more than it is a set of “cognitively effective” graphical symbols.

We consider that current notation-evaluation frameworks, such as PoN, represent a good starting point for notation analysis, but they are insufficient for creating models that readers understand. As argued by (van der Linden & Hadar, 2016), user-centric analysis of the visual notation of modeling methods complements PoN’s role of designing cognitively effective visual notations. Numerous visual notations have been evaluated with PoN. In addition, individuals have been involved in the design process of graphical symbols. However, little research shows their involvement in the design process of models that maintain the identity of the modeling notation.

Most visual notations are designed by IT modeling experts and remain difficult to comprehend by non-IT readers; for instance, (Woods & Bashroush, 2015) criticize notations such as ArchiMate for being “vertically optimized, limiting their attractiveness in many industrial projects”. In our research, readers not only evaluate visual notations but are also involved in suggesting improvements,

some of which are implemented in our models; for instance, they propose the metaphorical representation they find more suitable.

In evaluating the current SEAM notation, we need to consider the subjective nature of the PoN principles (van der Linden & Hadar, 2016). For instance, the semantic-transparency principle states that graphical symbols should suggest their meaning, but this meaning depends on the reader. We evaluated “how readers of the visual notation think, and what connotations they give to particular shapes and colors”, as suggested by (van der Linden & Hadar, 2016). By eliciting requirements from readers, modelers can find the “right balance” for applying the nine PoN principles for designing improved notations.

Appendix 6 Other IT Modeling Notations

We present below other modeling notations used in IT to model information systems.

Object-Process Methodology and Notation

OPM was designed for the specification of real-time systems (e.g., ATM machine). The below presentation of the OPM notation is based on (Dori, 2002).

OPM modelers design information systems, enterprises or technical products using objects and process models. OPM models are valuable in conceptual and preliminary design and serve as a decision support (Dori, 2002).

The OPM meta-model includes: objects, states, processes, specializations, aggregations, and result I/O links (Figure 115).

The OPM notation includes a concise set of symbols that enables the expression of the system's building blocks and how they relate to each other (Figure 115). OPM facilitates the symbolic representation of the objects of a system, its states, and the processes they enable. The objects are what a system or product is. Objects are put in classes without any perspective or viewpoint. The modeler defines objects to do certain actions without an organizational perspective or explanation of where objects and their names come from (similar to UML). The processes are what a system does. Connections are used to link objects and processes. OPM uses two types of elements: entities and links. Entities are used to express "physical" or "informational" elements, which can be inside or outside of the system being designed, i.e. "systemic" or "environmental".

OPM combines formal yet simple graphics with natural language sentences to express the function, structure and behavior of systems in an integrated, single model. It can be used to automatically generate program code and database schemas. The OPM designer chooses to show abstractions or zoom into some detail, for instance show how a specification migrates to implementation. Due to its intuitiveness, OPM is easy to communicate to stakeholders: peers, customers and implementers. The deliverables include a set of Object-Process Diagrams (OPDs) and a corresponding set of sentences written in the Object-Process Language (OPL).

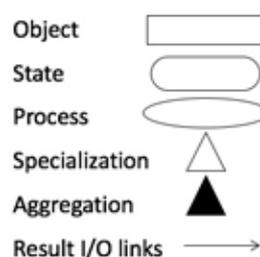


Figure 115: Object-Process Methodology notation – source: (Dori, 2002)

The OPM model below (Figure 116) shows two objects, a “Man” and a “Woman” that are combined into another object named “Couple”. A specialization of the “Couple” is object “Person” that has two states “single” and “married”. The process “Marrying” has as input a person that is “single” and yields a person that is “married” and a couple.

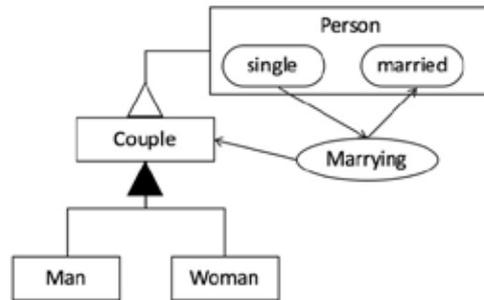


Figure 116: Object-Process Methodology model example – source: (Dori, 2002)

Design and Engineering Methodology for Organizations

DEMO was designed as a generic method for enterprise engineering.

DEMO modelers describe and explain the communicational dynamics and actions in organizations as well as how they should be (re)engineered (van Reijswoud, Mulder, & Dietz, 2001). They design business processes to obtain a detailed understanding of the reengineering and information infrastructure development that is consistent with the business requirements (Liu, Dietz, & Barjis, 2003). Models focus on both the social world and the object world, including the design of information systems through business process redesign.

In OPM, the essential concept is (business) transaction between the initiator and the executor. Transactions form business processes. Three principles are applied: information, action, organization. A transaction is composed of three phases:

- *Order phase*: two actors come to an agreement about the future execution of some action
- *Execution phase*: the negotiated action is executed
- *Result phase*: the actors negotiate the agreement about the result brought about in the execution phase

The operation of an organization is explained in terms of communicative actions between people. By means of the DEMO models it is possible to achieve a solid understanding of the types of transactions taking place in an organization, the participants involved in these transactions, the information that is needed and created during the transactions, and the relation between the different transaction types. The successful execution of a transaction in the social world (the world of communication) results in a change in the object world (the world of facts) in which the actors exist.

The speech act theory used in DEMO explains language as means of communication, which is the key concept for understanding and modeling organizations. Speech act considers the use of language as a form of rule-governed behavior. Uttering a sentence is the performance of an act, a so-called speech act.

There are a few assumptions made in DEMO:

- For the purpose of redesign and reengineering the business processes of an organization, one needs to have an understanding of its “construction” and “action”. Instead of applying the black-box model, DEMO applies a white-box model to understand organizations.
- Organizations belong to the category of social systems, meaning that the active elements are social individuals or subjects that behave according to assigned authority and corresponding responsibility against a common background of social norms and values.
- Information systems belong (only) to the category of rational systems, meaning that they do not make decisions, but only calculations, and in doing so only support decision-making.

DEMO models are compact and are usually done with pen / pencil on a single sheet of paper. The functioning of organizations is viewed from three levels:

- *Documental level*: an organization is regarded as a system of operators that produce, store, transport, and destroy documents

- *Informational level*: an organization is considered as a whole of processors that send and receive information, and perform calculations on this information in order to create derived information
- *Essential level*: an organization is considered as a networks of interrelated business transactions, which in turn are composed of interrelated communicative acts

The modeling facility of DEMO provides a graphical representation of the transactional structure of organizations. This transactional structure is represented in five partial models developed incrementally:

- *Interaction model*: shows the transaction types and the initiating and executing actors in an organization
- *Business process model*: shows the causal and conditional relations between the transaction types and its constituting phases
- *Fact model*: shows a complete specification of the state space of the object world
- *Interstruction model*: shows the actors and the information banks they need access to in order to execute a transaction
- *Action model*: shows the most detailed specification of the transaction structure of an organization

The example from Figure 117 includes two worlds: the social world and the object world. A sequence of transactions occurs in different phases between two elements, I and E. In the first phase, I requests and then E promises in the social world. In the execution phase the execution transaction occurs in the object world. In the results phase E states and I accepts leading to a fact in the object world.

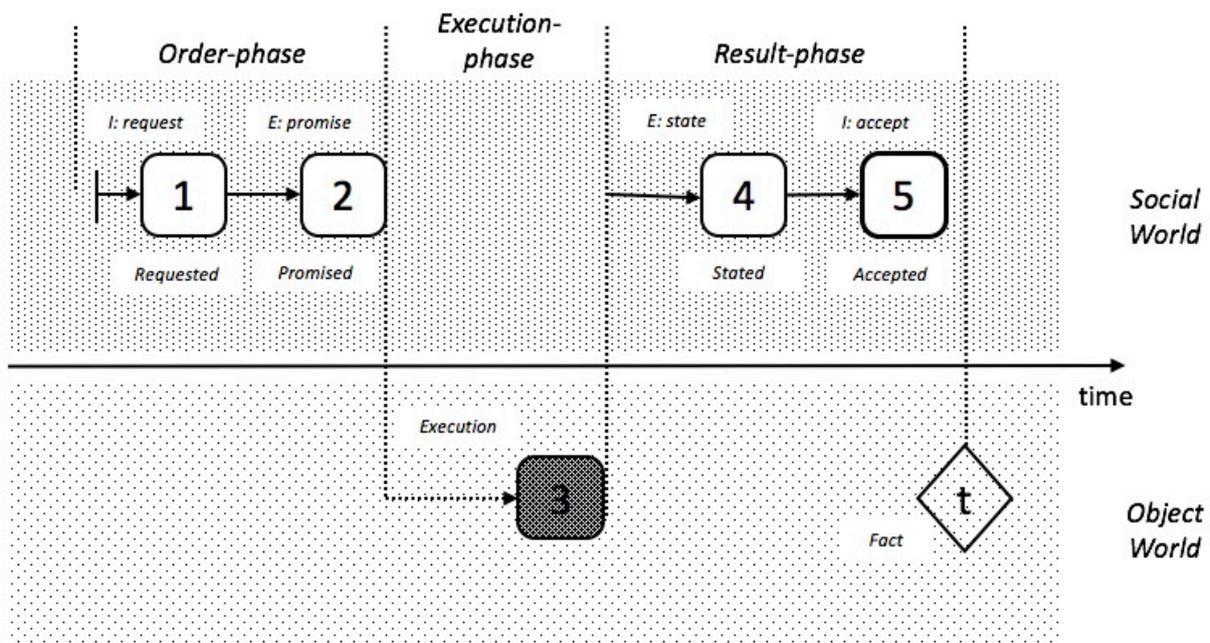


Figure 117: Design and Engineering Methodology for Organizations model example – source: (van Reijswoud, Mulder, & Dietz, 2001)

Soft Systems Methodology and Rich Pictures

SSM helps structure an exploration of a “situation” which some people regard as problematical (Checkland, 1990). Modelers do not follow a prescriptive process which has to be followed systematically but a framework to deal with “messy soft problem situations” which involve psychological social and cultural elements that lack a formal definition (Checkland, 1999). When modelling, people are usually unable to organize thoughts and expression consciously in several layers. Therefore, systems thinkers are adept at consciously separating “whether” from “what” and “how”. Soft Systems Thinking assumes the world is problematic, but the process of inquiry into the problematic situations that make up the world can be organized as a system. SSM models do not replace other models but enhance them.

By opposition, Hard Systems Methodology (HSM) modelers explore an “obvious” problem requiring a solution. Hard Systems Thinking assumes that the world is a set of systems that can be systematically engineered to achieve objectives, e.g., organizations seen as coordinated functional task systems seeking to achieve declared goals and that see the task of management as decision making in support of goal seeking. These models are useful in situations in which goals and measures of performance are clear-cut, communication between people are limited and prescribed, and in which the people in question are deferential towards the authority that set the goals and the ways in which they were to be achieved.

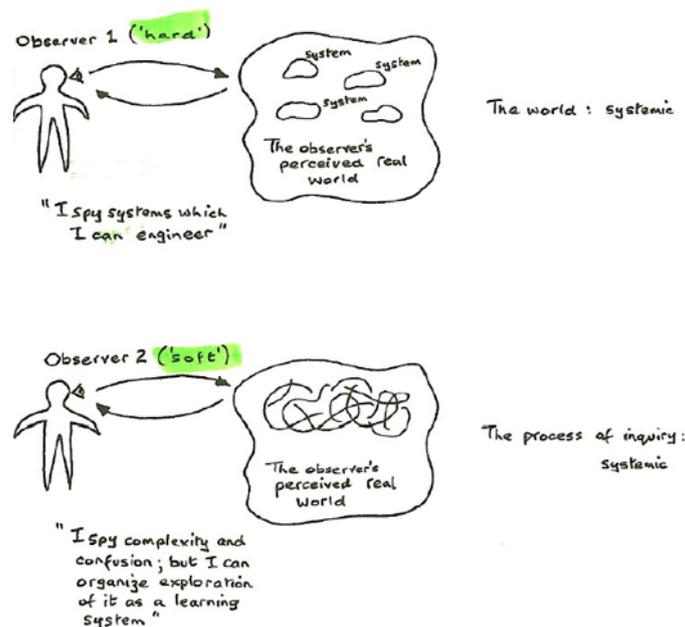


Figure 118: Hard and Soft Systems Methodology – source: (Checkland, 1990)

In SSM the notion of observer, and, as a consequence, his viewpoint, is critical. The observer gives an account of the world, or part of it, in systems terms: his purpose, his definition of his system(s), the principle(s) which makes) them coherent entities, the means and mechanism by which they tend to maintain their integrity, their boundaries, inputs, outputs, and components and their structure (Checkland, 1990).

SSM considers human activity systems as sets of linked activities which together could exhibit the emergent property of purposefulness. Before modeling can begin choices have to be made and de-

clared. It is necessary to decide for each selected purposeful activity the perspective or viewpoints from which the model will be built.

Models depict all primary stakeholders, their relations and their concerns. The main elements are based on the CATWOE framework: customers (C), actors (A), transformation process (T), world view (W), owner (O) and environmental constraints (E). There are 5 relevant characteristics of the model: efficacy (E1), efficiency (E2), effectiveness (E3), ethical (E4) and elegant (E5).

The Rich Pictures is the notation used for SSM. It is not precisely specified. The Rich Pictures model (Figure 118) is a cartoon-like representation, often done with pen and paper or as simple computer drawings, that identifies all the stakeholders, their concerns and some of the structure underlying the work context (Monk & Howard, 1998). It is a tool that can be incorporated in any design. SSM assumes that the model's complexity (number of elements) is due to the complexity of multiple relations. Rich Pictures draw attention to the many people or groups who could be seen as stakeholders in any human situation. The core of Rich Pictures is the understanding of human activity systems in a way that is meaningful for the actors in that system. Model views can be elicited on what people perceive to be important, but identifying and expressing multiple viewpoints of a work situation might be challenging.

Rich Pictures are typically constructed by interviewing people and are useful to be done at the workplace. Stakeholders participate in the process by working with the analyst to identify structures, processes and concerns significant to them. The model creation process is iterative and focuses on refined understanding through storyboarding.

A Rich Picture can be used as a first step in a lightweight design process to reason about the required redesign of the work (Monk & Howard, 1998). Rich Pictures are easy to learn and apply, but different people apply them in different ways (Wilson, 1990).

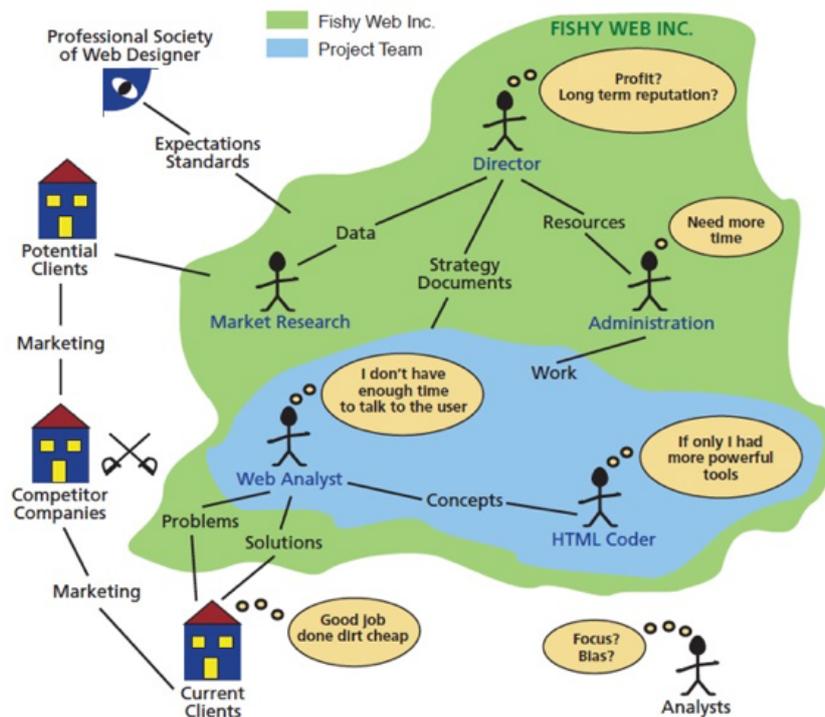


Figure 119: Soft Systems Methodology model example using the Rich Picture notation – source: (Monk & Howard, 1998)

The example from Figure 119 presents the internal and external environment of a company named Wishy Web Inc. All members of the company have roles. Some express certain concerns. For instance, the director thinks about the profit and the long term reputation of the company, the administration needs more time, the web analyst does not have enough time to talk to the user and the code hopes to have had more powerful tools. Actors are connected with resources, data, documents or concepts. Outside the organization several other actors are shown: the professional society of web designers that exchange standards with Fishy Web, potential clients, competitors, current clients with problems and solutions and analysts.

Business Process Modeling Notation

BPMN is used to show the flow of information between systems. BPMN 2.0 is an Object Management Group (OMG) international standard and one of the leading process modeling notations.

The meta-model includes objects, activities, states, gateways and sequences.

The BPMN notation is similar to UML and UCM, and relies on software engineering practices (Figure 120).

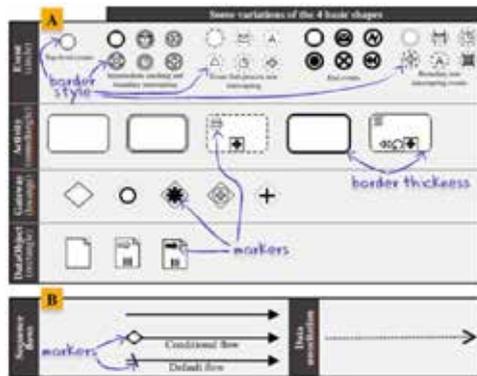


Figure 120: Business Process Modeling Notation notation– source: (ConceptDraw, 2016)

The example from Figure 121 is taken from (ConceptDraw, 2016), a software for drawing business process models based on BPMN 2.0. It shows a cab booking process between a travel agent, a cab driver and a customer. First the customer requests a booking. In order to receive the booking details, the travel agent needs to get the booking request, then check availability, get alternative time and propose booking status. The booking is confirmed both to the customer and to the cab driver. The driver will pick up the customer and complete the assignment by notifying the agent. The process ends for both the travel agent and the cab driver.

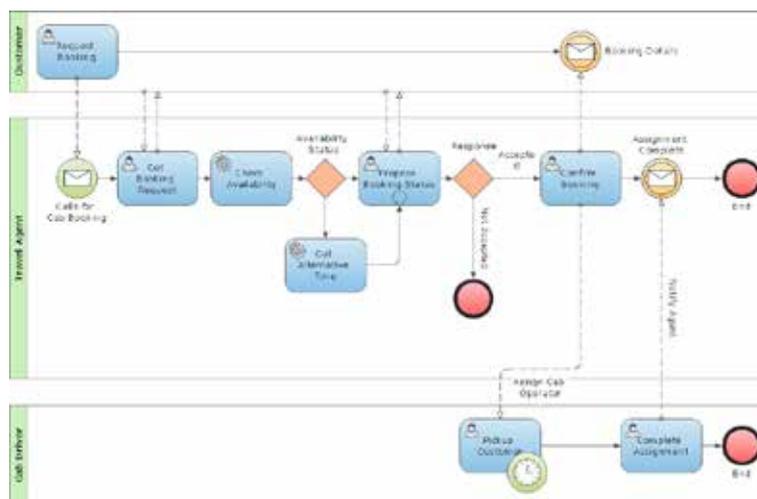


Figure 121: Business Process Modeling Notation model example - source: (ConceptDraw, 2016)

ArchiMate

ArchiMate was designed to integrate UML for enterprise modeling. It is an open and independent international standard method that follows the terms used by the Open Group Architecture framework (TOGAF) (Lankhorst M. M., 2013). TOGAF is the most widely used enterprise architecture (EA) framework and methodology. It is an international standard available for internal use by any organization. ArchiMate provides an architectural approach that enables visualization of different domains and the underlying relations and dependencies. It contains meta-models for different levels of specialization (Groenewegen, Hoppenbrouwers, & Proper, 2010). There is a generic meta-model that explains the overall concepts while the specific ones explain the individual architectural domains. With ArchiMate and TOGAF modelers develop a pyramidal view of the enterprise (metaphor shown Figure 122), which starts from the vision and strategy defined by the management and go down to key performance indicators and actions performed by workers. Multiple levels can be identified - Level1: Vision, values, mission; Level2: Goals and objectives; Level3: Actions and approaches and Level4: People, systems and resources (Figure 122). The four levels of analysis (business, application, technology and infrastructure) include structure, behavior and information in an integrated way via a succession of internal and external views of systems, from business to IT (Jonkers, Lankhorst, & Proper, 2011).

ArchiMate modelers want to improve the efficiency within organizations via industry consensus and strategic management practices. These practices view the enterprise as a machine with fine-tuned parameters in which executives set the strategy and objectives are clear-cut, derived from this strategy and refined into IT architecture. All stakeholders follow duly the objectives and know their mission.

ArchiMate uses the TOGAF meta-model. It is based on the vocabulary of different disciplines; thus, it is extensive. TOGAF designers model generic elements by thinking in a generic way and specific elements by applying various disciplines.

The ArchiMate notation is presented in Figure 122. It includes numerous graphical symbols.

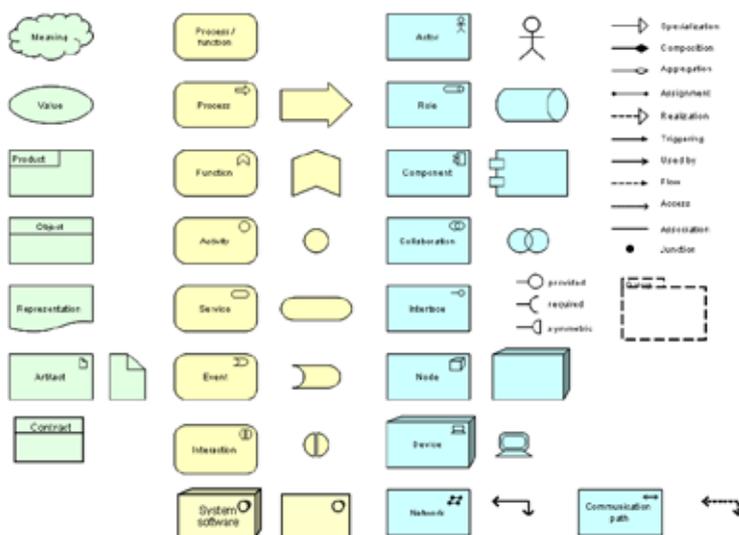


Figure 122: ArchiMate notation – source: (The Open Group, 2016)

The example from Figure 123 shows the operation of an insurance company. At the top, the business layer includes a top level detailing four roles and actors: a client, an insurant, ArchiSurance and an insurer, a middle level with external business services: claim registration service, customer information service and claims payment service and a bottom layer with damage claiming process: registration, acceptance, valuation and payment. The following two levels for the middle, application layer, which includes an external application services level including claims administration service, customer administration service, risk assessment service and payment service. Then, application components and services level includes claims administration, customer administration, claim information service, risk assessment and financial application. Finally, the third, infrastructure layer, includes two levels: external infrastructure services includes claim files service and customer files service and infrastructure that includes a zSeries mainframe with a DB2 database, a Sun Blade with an iPlanet app server and a Risk assessment EJB.

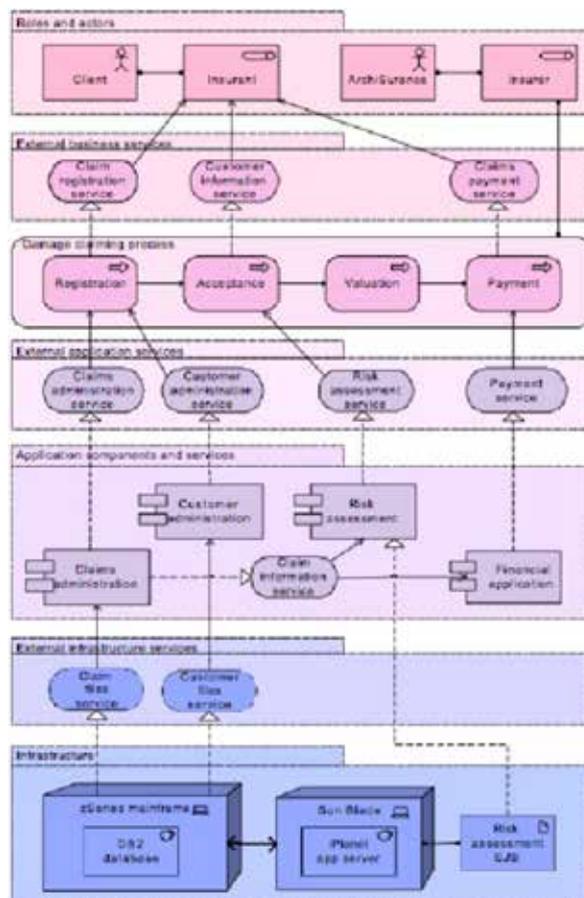


Figure 123: ArchiMate model example – source: (Kraan, 2016)

The ArchiMate notation is richer than SEAM. ArchiMate is more difficult to be used in workshops as models include seven modeling levels whereas in SEAM models there are usually three levels: the market segment as a composite, the enterprise as a composite and the enterprise IT as a composite. The principle in ArchiMate is to align boxes near other boxes to model levels one on top of the other whereas in SEAM boxes are placed many (wholes) inside one (composite). In an ArchiMate model, elements can be connected at any level in the hierarchy whereas in SEAM the connection is established with the upper or lower level online. Similar to UML, the context in ArchiMate is not important and therefore not shown, whereas in SEAM is the system as a composite present in every model. ArchiMate focuses only on the solution without the problem, as it is the case with SEAM.

Use Case Map

UCM modelers describe functional requirements of a system as causal scenarios (Buhr & Casselman, 1996).

The UCM meta-model includes: team, process, object, agent, actor, component, link.

The UCM notation has few entities such as components (team, process, object, agent and actor), protected components and context-dependent components and numerous connections such as paths, responsibility, empty point, direction arrow, waiting place, empty point, direction arrow, etc. (Figure 126) (Amyot, 2003).

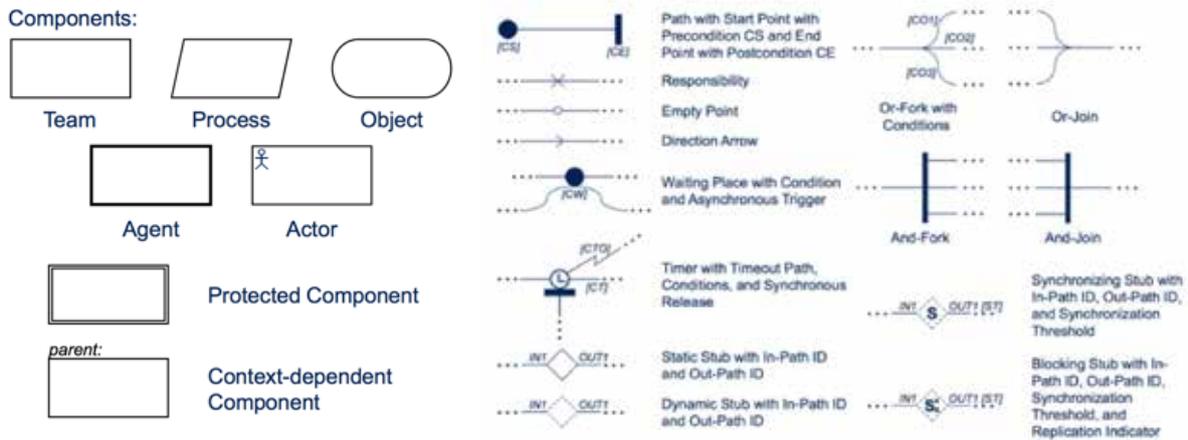


Figure 126: Use Case Map notation

The brief example from Figure 127 is taken from the Model-Based Design and Verification of Distributed Real-Time Systems at the University of Ottawa by professor Gregor Bochmann (Bochmann, 2016). It shows the relations between three components: a warehouse, an office and a client. First, the office receives an order. If the order is rejected, then it will later be closed and the process ends. If the order is accepted, then it will be filled. In this case the warehouse will ship the order and the invoice will be sent to the client. The client makes the payment and the office accepts it. The order is then closed by the office.

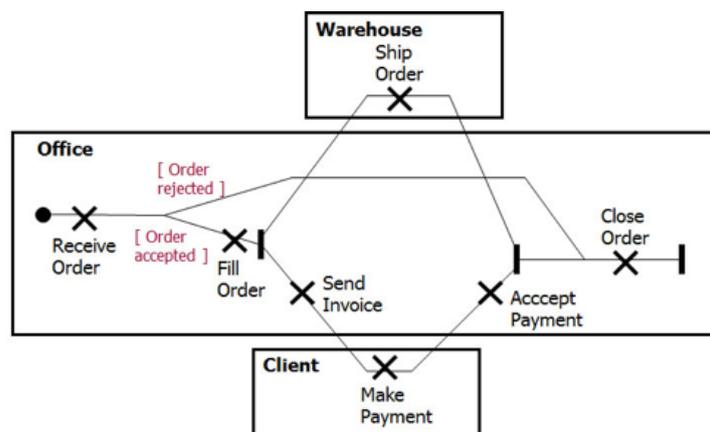


Figure 127: Use Case Map model example

e³-value

e3-value modelers conceptualize and visualize a business idea, starting from understanding which enterprises and actors are involved, to an assessment of profitability for each enterprise. Businesses are modeled as networks of enterprises. Modelers focus on creating high-level business models, without going into details about the processes and technologies. They want that stakeholders involved in the business model reach a better understanding of it. Based on this, they do an analysis and profitability assessment of the business model for all stakeholders.

The e3-value meta-model includes: actors, values and dependencies (Figure 128).

e³-value models show the flow of value between different actors. Value exchanges interconnect value ports of actors or values interfaces (Figure 128). e³-value is a lightweight and traceable but lacks marketing perspective.

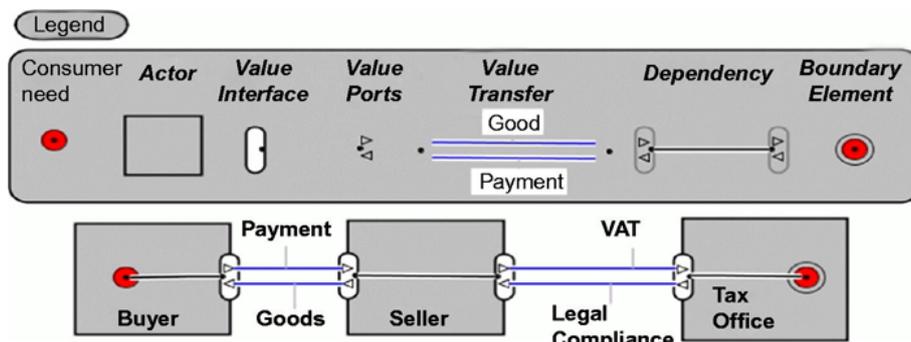


Figure 128: e³-value notation

The example from Figure 129 is taken from (Bertrand & Schmitt, 2006). It shows a brokerage firm operating in the tourism sector. There are three parties involved: a customer, a broker and a provider. The broker operates between customers and service providers. Customers contact the broker who matches its requirements with the service offerings of tourism service providers. Regarding payment, he can receive it from customers on behalf of the providers. The providers transfer the broker the right to book their products and services and pay the broker a commission (awareness fee) for the mediation of their services. The service provider delivers the products and services bought by the medium of the broker to the customer. Alternatively, the customer can benefit from the tourist service directly from the provider.

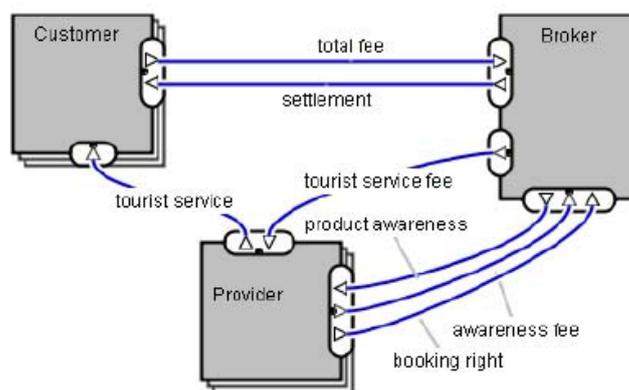


Figure 129: e³-value model example

Appendix 7 Preliminary Study on Service Modeling

As mentioned in Chapter 3.4, we performed a preliminary study in which we interviewed seventy-six participants during 2014-2015. Interviews were structured as informal discussions that lasted between 30 and 60 minutes per participant. They took place at the Ecole Polytechnique Fédérale de Lausanne, at the interviewees' homes, their workplaces, or in public places. Participants were (undergraduate and PhD) students, secretaries, analysts, managers, doctors, lawyers and economists.

Each participant was asked to create one hierarchical model and one service model of their organization using only pen and paper. At the end of the model creation, each participant explained the two models to the interviewer (thesis author), who took notes. The interviewer collected one hundred and fifty-two models created by all participants. The models represented the interviewee's perceptions of their organizations. For each participant, the modeler also collected information about his age, nationality, sector of activity, organization, position, gender, name and profile (IT or Business).

First, we present some demographical information about the participants (Figure 130). In terms of gender, 37 participants were male and 39 were female. In terms of profile, 38 had a predominant IT background, whereas the other half had a predominant business background. In terms of sector, the following were represented: research, administration, software development, consulting and finance. A large proportion of male participants work in research, software development and consulting, whereas a large proportion of female participants work in administration, finance and other sectors.

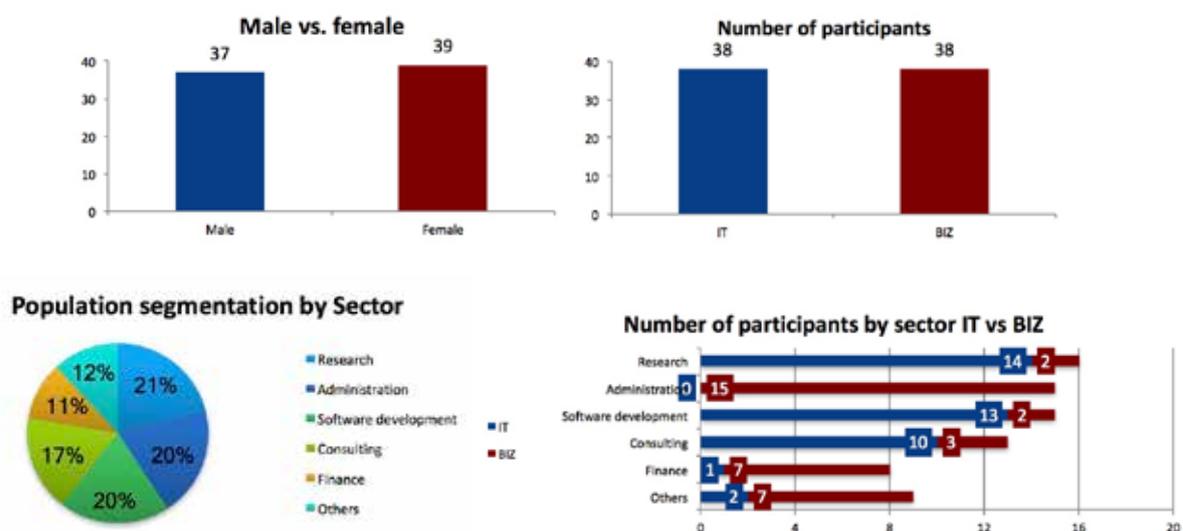


Figure 130: Demographics analysis of the preliminary study

Next, we present a summary of the results of this study. Analyzing hierarchical models was not interesting for our purposes because the SEAM notation is not used for this. In addition, our participants had a good understand of a what org charts are and they represented them well. We asked them to create a hierarchical model first in order to have them think of their role in their organization.

For each service model created by participants, we analyzed three main elements as follows:

1. Structure
 - a. Service organization: self, department and / or whole company
 - b. Topology: ontology, tree diagram, supply-chain
 - c. Distance and closeness: distance between figures (relative to the model; can be small, medium and large)
 - d. Compartmentalization: if figures are placed inside compartments
 - e. Encapsulation: if one figure is enclosed and separated from others
 - f. Barrier: if two or more figures are separated by lines
2. Content
 - a. Composition: suppliers, supplier partners, competitors, competitor partners, regulation authorities, customers, customer partners, manager
 - b. Resources and needs: money, technology – information – ideas, physical objects and tools, competencies and capabilities
 - c. Interactions and views: named interaction, unnamed interaction, conflicts and compromise, constraints, goals – beliefs, internal view, external view
 - d. Actions / services / processes: shared actions, offered / received actions, sub / supra actions, chronology / delays, flow / input-output
3. Notation
 - a. Shapes: rectangles (actors: people / organizations, or services), ovals (actors: people / organizations, or services), specific shapes (people / organizations), text (actors: people / organizations, or services), simple arrows (no, one or two directions / dashed or not), double arrows (no, one or two directions, dashed or not), boundaries (dashed or not)
 - b. Position: self (top, center, left, bottom, right), project-team / leader (top, center, left, bottom), customer (top, center, left, bottom, right)

We present the results of our analysis of an “average” service model created by our participants.

Regarding structure, service modeled include the participants’ company (82%), their department (12%), and them-selves (7%). In terms of topology, ontology is preferred (54% of models), followed by a tree diagram (24%) and supply chain (22%). Participants used mostly encapsulation (68%), compartments (21%) and barriers (12%).

Regarding content, participants represented an average number of 11 connections among 13 entities (Figure 131). There are minor differences between IT and business people. IT people tend to create simpler models, with a lower number of entities and connections. The average number of actors (among entities) is 7 for IT participants and 6.5 for business participants. These actors include 2 proper nouns and 5 common nouns (on average): customers (67%), suppliers (58%) and management (34%). Besides actors, participants showed on average 3 resources: information (57%), physi-

cal objects (46%), competencies (59%) and 3 actions: un-named (79%) and named (24%). Internal views were used in 76% of models whereas external views in 45% of them.

There are numerous similarities regarding the concepts included by IT and business people in their models (Figure 132). Most concepts are common nouns that are used to denote resources and actions. The context is seldom shown. Service models include an average of 5.5 common nouns and 2 proper nouns. IT people represent less resources and more actions.

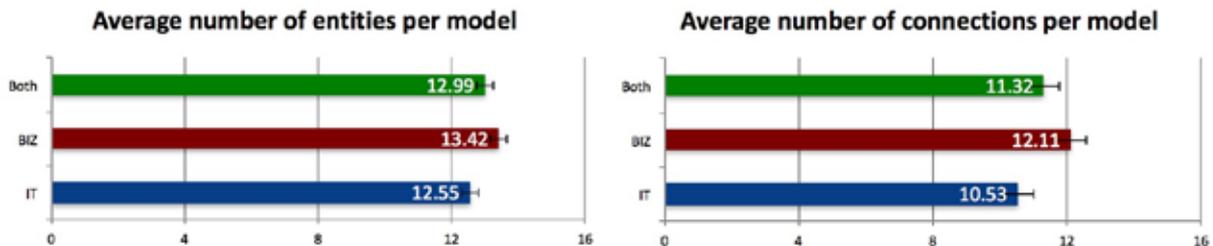


Figure 131: Average number of entities and connections between entities in participants' services models

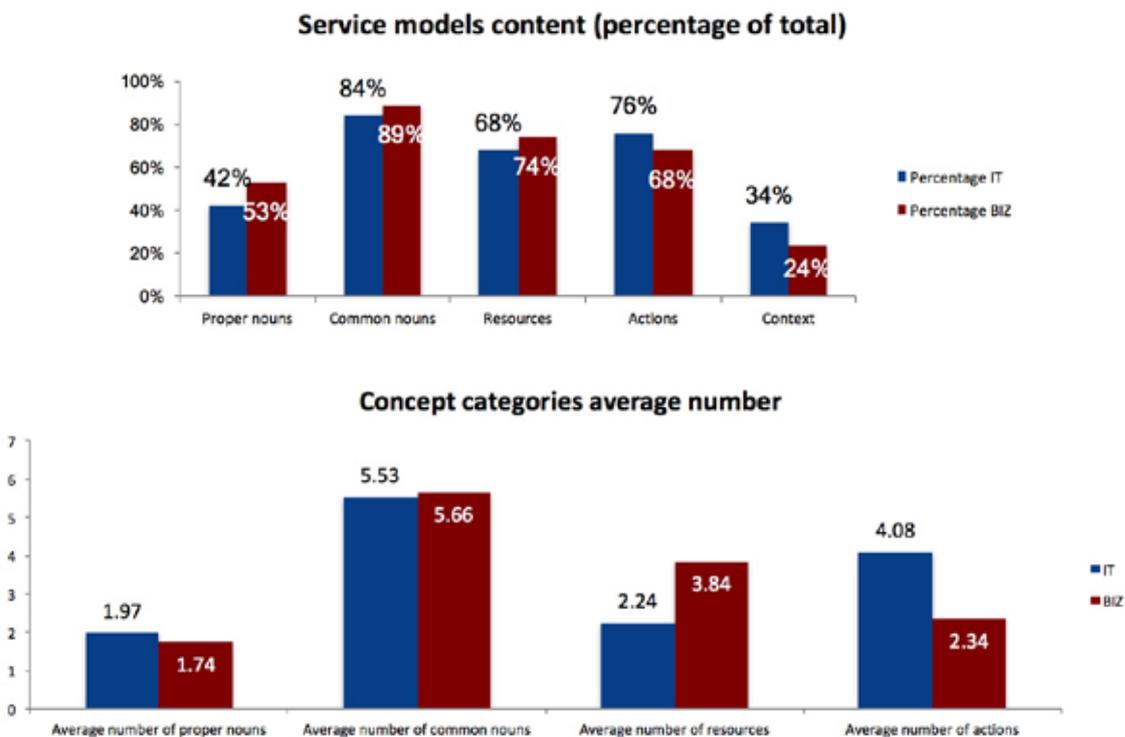


Figure 132: Concept analysis of participants' service models

Regarding the notation, participants used abstract shapes: rectangles (55%) and ovals (21%) for organizations, more than for people and services; text is equally used for organizations (26%), services (25%) and people (24%); unidirectional lines (72%), borders (32%), simple lines (25%) and bidirectional lines (17%) are also used. The self is represented in the center (24%), to the left (18%) or at the top (11%), the management at the top (38%) or in the center (12%), and the customer to the right (36%), at the bottom (20%), or at the top (12%).

We compared the models created by business and IT people in terms of the model's structure, content and notation. The main differences are synthesized in Table 14. For structure, ontology is more preferred by IT people (61%), whereas tree diagrams are more preferred by business people (32%).

In terms of content, business people represent more resources, money and actions, whereas IT people represent more technology and physical objects (the found significant statistical differences only for resources and actions). Regarding notation, business people use more text to depict organizations and position the management in the center, whereas IT people use more borders.

	Criteria	BIZ	IT	Significance
1. Structure	Ontology	47%	61%	
	Tree diagram	32%	16%	
2. Content	# resources	3.8	2.2	P = 0.025
	# actions	4.1	2.3	P = 0.020
	Money	24%	5%	
	Technology	50%	65%	
	Physical objects	35%	60%	
3. Notation	Text for organizations	34%	18%	
	Borders	16%	47%	
	Management: center	21%	3%	

Table 14: Summary of main differences between business and IT participants' models

We also compared the models created by SEAM practitioners against the service models created by our participants. We analysed twenty SEAM models created by experts. An “average” SEAM service model has 3 hierarchical levels (wholes and composites) and contains 24 connections among 23 entities. Out of these 19 are actors (7 human and 12 non-human), 4 resources, 13 services (wholes, no composite), 8 processes (1 as composite). This model uses 6 shapes and an average of 3.5 colors. Therefore, an “average” SEAM model with one hierarchical level (context) has 3 times less entities and connections: 7.5 connections and 7 entities: 6 actors (2.5 human and 3.5 non-human), 1 resource, 3.5 services and 2.5 processes. We conclude that a 2-level “average” SEAM model is similar to an “average” model created by participants.

SEAM expert modelers' and participants' models share the following similarities:

- They combine internal and external views showing the company, the department and the self
- They include a customers, suppliers and management
- They include actors that share the similar positioning: customers (right), suppliers (left), management (top), self (left or center)

The differences between the participants' models and the experts' models are:

- Participants prefer modeling one context at a time (max. 2 hierarchical levels)
- Oriented lines are preferred by participants and not used in SEAM
- Unnamed are more frequently used than named actions (hide services for some actors)
- Participants use abstract shapes but they found more useful the use of pictograms

In conclusion, the service models created by participants are based on implicit suppositions. SEAM can make them more explicit.

We applied the Physics of Notations to evaluate each of the models against each principle (Figure 133). Models satisfy well the last principles: graphic economy and cognitive fit, but not so well the semiotic clarity, the perceptual discriminability and the semantic transparency.

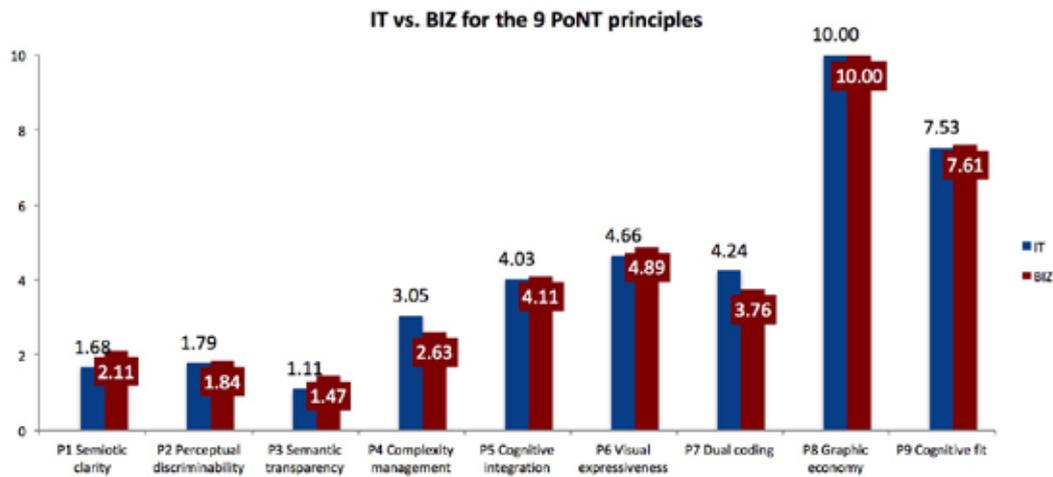


Figure 133: Evaluation of participants' service models with the Physics of Notations principles

Curriculum Vitae

CONTACT INFORMATION

- Rue de Carouge 69, 1205, Geneva, Switzerland
- +41 76 269 54 31
- george.popescu@epfl.ch / popescugeorge2003@yahoo.com
- popescugeorge.com / linkedin.com/in/popescugeorge

PERSONAL INFORMATION

- 18th of April 1986
- Married

PROFESSIONAL EXPERIENCE

Ecole Polytechnique Fédérale de Lausanne (EPFL)

Research assistant

Switzerland
2010-2016

- Conducted research in the field of business / IT alignment: developed guidelines for modelers to design models that are understandable by readers based on 120 field interviews
- Taught 9 courses in Business and Technology to 350 Bachelor and Master students
- Supervised 15 undergraduate students in the field of service design, visualization and requirements

Siemens Wind Power

IT Applications Intern

Denmark
2010

- Established an enterprise-wide SAP e-learning platform adapted for management presentations used by 6k employees
- Led a company-wide software usage optimization project resulting in 40% cost reduction by removing unused licenses
- Defined 10 key performance indicators for the IT department to ensure daily automated service reporting to management

Oracle Corporation

Technology Presales Consultant

Romania
2008-2009

- Contributed to 10 cross product requests for proposals from large UK clients (€5M revenue) by performing qualitative and quantitative research on vendor proposals for cost, technology approach, delivery date and terms of service
- Designed and released the marketing newsletter for the UK division resulting in 5 new large clients (10k employees)
- Managed the allocation of 20 senior consultants to 8 major client engagements resulting in 30% scheduling efficiencies

Orange

IT Networking Trainee

Romania
2007

- Ensured 99.99% organization-wide network service availability

EDUCATION

Ecole Polytechnique Fédérale de Lausanne (EPFL)

Doctorate in Computer Science, specialization Business Informatics (GPA: 5.33 / 6)

Switzerland
2010-2016

Ecole Polytechnique ParisTech and INSTN Université Paris-Sud 11

Master of Science in Conception and Management of Complex Information Systems (GPA: 14 / 20)

France
2009-2010

Polytechnics University of Bucharest

Bachelor of Science in Systems Engineering, specialization Applied Informatics (GPA: 8.94 / 10)

Romania
2005-2009

National College Anastasescu
International Baccalaureate (9.91 / 10), specialization in Mathematics, Informatics, English (GPA: 10 / 10)

Romania
2005-2008

AWARDS

Recognized in the top 6 post-graduate Romanian students in Europe by The League of Romanian Students 2014
Evaluated with an IQ test score of 135 corresponding to the top 1% percentile by Mensa Switzerland 2011
Awarded 1st prize for an autonomous robot specification and design by Ecole Polytechnique ParisTech 2010
Awarded Masters excellence scholarship (top 5%) by Ecole Polytechnique ParisTech 2009
Awarded 1st prize for a Bluetooth robot voice control software by the Polytechnics University of Bucharest 2009
Awarded 1st prize for a new image color-coding algorithm by the Polytechnics University of Bucharest 2006
Awarded Bachelors merit scholarships (top 5%) by the Polytechnics University of Bucharest 2005-2009

LANGUAGES

English (fluent, TOEFL), **French** (fluent, DELF), **German** (advanced, Goethe), **Romanian** (native)

EXTRACURRICULAR ACTIVITIES

Leadership President of the Doctoral Commission at EPFL (3 years) conducted 3 meetings with the management on assistants' salaries based on 400 responses and presented campus information to 60 new PhDs quarterly
President of the Romanian Students Association at EPFL (2 years) lead a 200 members cross-functional team
Consulting External consultant for 30 business improvement and IT optimization projects totaling €5M at 15 small and medium organizations

INTERESTS

Tennis Winner of 60 singles and doubles competitions in Switzerland, Denmark, France and Romania

PUBLICATIONS

Popescu, G. and Wegmann, A. *Using the Physics of Notations Theory to Evaluate the Visual Notation of SEAM*. IEEE 16th Conference on Business Informatics. Geneva, Switzerland, 2014.

Popescu, G., Tapandjieva, G. and Wegmann, A. *Business and IT design with SEAM: an Illustration with the PhD Hiring Process at École Polytechnique Fédérale de Lausanne*. IEEE International Conference on Systems, Man, and Cybernetics. Manchester, United Kingdom, October 2013.

Regev, G., Bajić-Bizumić, B., Golnam, A., Popescu, G., Tapandjieva, G., Saxena, A. B. and Wegmann, A. *A Philosophical Foundation for Business and IT Alignment in Enterprise Architecture with the Example of SEAM*. Third International Symposium on Business Modeling and Software Design. Noordwijkerhout, Netherlands, July 2013.

Wegmann, A., Bajić-Bizumić, B., Golnam, A., Popescu, G., Tapandjieva, G., Saxena, A. B., Yassaee, M. and Regev, G. *Requirements Modeling in SEAM: The Example of a Car Crash Management System*. Comparing Requirements Modeling Approaches Workshop, Requirements Modeling Conference. Rio de Janeiro, Brazil, July 2013.

Popescu, G. *Group Recommender Systems as a Voting Problem*. HCI International, Springer. Las Vegas, NV, USA, July 2013.

Popescu, G. *Adaptive Voting Algorithms for Group and Social Recommender Systems*. HCI International, Springer. Las Vegas, NV, USA, July 2013.

Popescu, G. *Designing a Voting Mechanism in the GroupFun Music Recommender System*. HCI International, Springer. Las Vegas, NV, USA, July 2013.

Popescu, G. *Eye Tracking Group Influence - Experiment Design and Results*. Technical Report. École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, September 2012.

Popescu, G. and Pu, P. *What's the Best Music You Have? Designing Music Recommendation for Group Enjoyment in GroupFun*. ACM SIGCHI Conference on Human Factors in Computing Systems, Works-in-Progress, Austin, Texas, USA, May 2012.

- Popescu, G. and Pu, P. *A Different View on Group Analysis - (Eye-) Tracking Social Influence*. Technical Report. École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, January 2012.
- Popescu, G. and Pu, P. *Probabilistic Game Theoretic Algorithms for Group Recommender Systems*. ACM RecSys, Workshop on Music Recommendation and Discovery, Chicago, IL, October 2011.
- Popescu, G. *Validation of Game Theoretic Algorithms for Recommender Systems*. Semester report. École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, July 2011.
- Popescu, G. *Group Recommender Systems as a Voting Problem*. Semester report. École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, February 2011.
- Popescu, G. *Management of IT Applications at Siemens Wind Power*. Master thesis. École Polytechnique, Paris, France, October 2010.
- Popescu, G., Fang, Z., Gourdol, J.-N., Haas, J., Indre, M., Kusteiko, D., Kusteiko, V., Perciu-Aldea, A., Pavel, A. M., Spasova, I., Topa, A.-C. and Boukdhir, W. *Robot Challenge Requirements Analysis and Design*. École Polytechnique, Student Scientific Communication Session 1st Prize, Paris, France, March 2010.
- Popescu, G. *Voice Command Kephera III*. Bachelor thesis. Polytechnics University of Bucharest, Bucharest, Romania, July 2009.
- Popescu, G. *Voice Command Kephera III*. Student Scientific Communication Session 1st Prize. Polytechnics University of Bucharest, Bucharest, Romania, May 2009.
- Popescu, G. and Bălan, C. *Contributions on Knowledge Management for Cooperation Agent Systems*. Student Scientific Communication Session 3rd Prize. Polytechnics University of Bucharest, Bucharest, Romania, May 2008.
- Popescu, G. *Event Methods for Traffic Management*. Student Scientific Communication Session 4th Prize. Polytechnics University of Bucharest, Bucharest, Romania, May 2007.
- Popescu, G. and Bălan, C. *Adaptive System for Traffic Management*. Student Scientific Communication Session 1st Mention. Polytechnics University of Bucharest, Bucharest, Romania, May 2007.
- Popescu, G. *Coding of Forms*. Student Scientific Communication Session Special Prize. Polytechnics University of Bucharest, Bucharest, Romania, May 2007.
- Popescu, G. *Binary Coding of Images*. Student Scientific Communication Session 1st Prize. Polytechnics University of Bucharest, Bucharest, Romania, May 2006.