

A Catalogue of iStar Extensions

Enyo Gonçalves^{1,2}, Tiago Heineck³, João Araújo⁴ and Jaelson Castro²

¹ Universidade Federal do Ceará, Campus Quixadá, Brazil

² LER, Universidade Federal de Pernambuco, Brazil

³ Instituto Federal Catarinense, Brazil

⁴ NOVA-LINCS, Universidade Nova de Lisboa, Portugal

ejtg@cin.ufpe.br, th@cin.ufpe.br, joao.araujo@fct.unl.pt,
jbc@cin.ufpe.br

Abstract. iStar has been extended since its initial proposal in the 90's. It is worth noting that since 2016 the language notation is under standardisation. However, new extensions continue to be proposed. The search for previous extensions and its constructs can be a starting point in the proposal of new ones. So, to ease the identification of the previous extensions and its constructs is essential to next proposals. Motivated by this situation, the objective of this paper is to present a catalogue of iStar extensions. The extensions and their constructs were identified from a systematic literature review, which identified that 96 extensions had been proposed until 2016. The results suggest that catalogue is important during future iStar extensions proposal.

Keywords: Requirements, Catalogue, iStar extensions.

1 Introduction

iStar is a goal-based modelling language used to model requirements at early and late phases of software development. Since its proposal by Yu [19], the language is often extended to incorporate new constructs related to an application area. Therefore, we performed a Systematic Literature Review (SLR) [9] which identified 96 extensions proposed until 2016. The iStar community widely uses part of the extensions to model systems or as the basis for other extensions. We can cite TROPOS [3], Goal Requirements Language (GRL) [2] and Secure Tropos [8] as examples of extensions widely used as the basis for the proposal of other extensions. Consequently, to find a specific iStar extension based on its characteristics or find a specific construct are tasks that require extra time and can be unsuccessful.

Due to the proposal for a new version of iStar [6], we believe this is the best moment to discuss how iStar extensions could be systematized. We are interested in improving the way of extending it. Thus, we intend to define a software process, so the definition of a catalogue of iStar extensions and its constructs is an interesting step in this direction. The experts in iStar extensions suggested in a qualitative study [10]

that a catalogue of iStar extensions could be useful for extenders during the proposal of new iStar extensions.

Motivated by this scenario, this paper aims to present a catalogue of iStar extensions, including its constructs and analysis about both. This catalogue is important to ease the identification by the extenders of the existing extensions and constructs previously proposed, and its reuses. The catalogue is based on the results of our SLR [9] and the mitigation of conflicts [11] which established new representations and prioritisation of the constructs with conflict. It is available at <http://istarextensions.cin.ufpe.br/catalogue/>.

This paper is organised as follows. In Section 2, the background is described. In Section 3, the related work is presented. Section 4 shows the methodology used to guide this proposal. The preliminary results are presented in Section 5. Section 6 presents the CATIE: A Catalogue of iStar Extensions is detailed. And finally, Section 7 shows the conclusions and future work.

2 Background about iStar Extensions

There are different forms to present an iStar extension [9], but all of them introduce new concepts to iStar. For example, a set of extensions described in detail the new concepts and its representations in the iStar metamodel and concrete syntax (see works of Ali, Dalpiaz and Giorgini [1] and Morandini et al. presented in [14]). This kind of extensions describes how the new concepts were introduced and how to use them.

On the other hand, a set of other extensions was presented as a method to create models, and the iStar changes are presented using illustrations with the usage of new concepts. An Example of this kind of extension is the extension of Islam et al. available at [12].

A set of papers presents a case study or a modelling tool with a set of new concepts introduced in iStar, for example in Gans et al. [7] and Siena et al. [17]. They were selected because they are the only evidence for these extensions.

We did not consider as an extension any work that used iStar without changes in abstract syntax (changes in metamodel or validation rules) or concrete syntax (new graphical representation) because in this case iStar is used with default syntax without any changes (extension).

TROPOS4AS is an example of iStar extension, it was proposed by Morandini et al. in [14]. It models characteristics of adaptive systems in Tropos, proposed by Bresciani et al. [3], providing conceptual models, a graphical language, and its semantics, to enable capturing requirements needed for defining and driving adaptation.

Fig. 1 shows an example of modelling with TROPOS4AS for a cleaner agent that needs to clean rooms. We can identify elements that are not part of iStar default syntax such as failure, error, condition, inhibits relationship, failure relationship and goal types. We highlight the goal types with the values of AchieveGoal, MaintainGoal and PerformGoal and the new graphical representations to condition and failure.

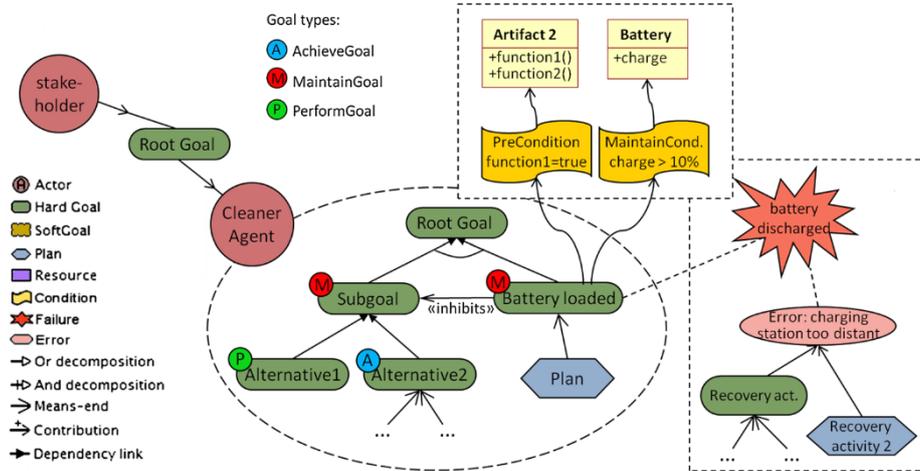


Fig. 1. Modelling of cleaner agent with TROPOS4AS[14]

3 Related Work

We did not find a catalogue of iStar extensions. However, catalogues have been proposed to contribute to join the knowledge of other aspects of the requirements engineering area.

Next, we present some papers related to catalogues of requirements. In [16] the authors present a gamification requirements catalogue for educational software including the personality types of students. This catalogue is based on the educational requirements identified by an SLR which were selected by experienced participants of a survey. Thus, the main result is the catalogue of educational requirements with 232 items which are grouped in 3 levels (dynamics, mechanics and components).

A catalogue of Functional Software Requirement Patterns (F-SRP) was presented in [15]. Specifically, the F-SRP in the catalogue addresses the domain of Content Management Systems (CMS). An SRP basically consists of: a template that generates one or more requirements; some information to identify its adequacy to a particular project; and how it may be tailored to that project. The catalogue includes 29 Non-Functional SRP and 37 Non-Technical SRP to an existing catalogue proposed previously by the same authors.

Finally, a reusable catalogue of legal requirements derived from specific legal texts regarding security and personal data protection was proposed by Toval et al. [18]. The Personal Data Protection (PDP) Catalogue enables requirements engineers to incorporate legal requirements into the development lifecycle and build compliance into new systems [18]. By providing reusable legal requirements, analysts can more easily uncover ambiguities and inconsistencies, and the quality of the catalogue increases the more it is used [18].

4 Methodology

The CATIE catalogue is based on the information resulting from an SLR about iStar extensions and the results of a study which involved an experiment and a survey for mitigating conflicts identified in the SLR.

Therefore, we specified the information to be extracted from the iStar extensions identified by the SLR and their constructs. We created a template to be used during the extraction. The template was used to extract the information of the selected papers of the SLR and their constructs. Finally, the CATIE was created and the extracted data was inserted into it. Fig. 2 summarises these steps followed to create the CATIE catalogue.

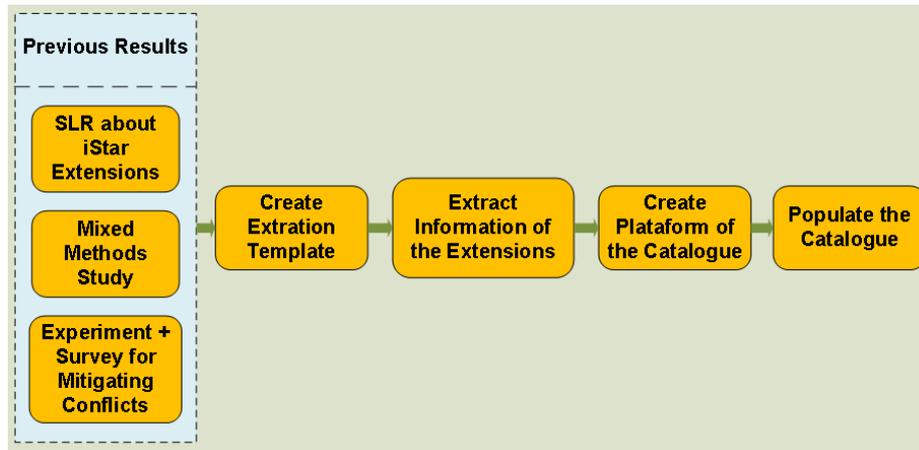


Fig. 2. The method used to create the catalogue of iStar extensions.

5 Base Results

Our catalogue is part of a research which is investigating how the iStar extensions have been proposed and what can be done to improve future iStar extensions. It based on our previous results. Therefore, these previous results are presented in this section, i.e. the SLR about iStar extensions and the studies for mitigating conflicts are briefly presented.

5.1 SLR about iStar Extensions

The SLR about iStar extension [9] identified the papers which propose iStar extensions. These papers were used in the step of extraction during the creation of the catalogue. The search was done in ACM, EI Compendex, IEEE Xplore, Science direct, Scopus, ISI Web of Science and Springer databases using the defined search string. The selection was based on four predefined inclusion and exclusion criteria, we also

realised snowballing and asked the specialists about the suggestion of papers. 96 iStar extensions were selected, and data were extracted to answer nine research questions. The research questions were proposed based on Model-Driven Development (MDD) concepts and the principle of semiotic clarity, which is used by Moody [13]. We also identified the main conferences and journals, authors and kind of validation.

Fig. 3 shows the publication frequency by showing the number of publications over time. Fig. 3 shows iStar extensions still being a field of interest: a greater amount of extensions was carried out in the last six years of the analysed period with approximately 61%. The results of this SLR are available in [9].

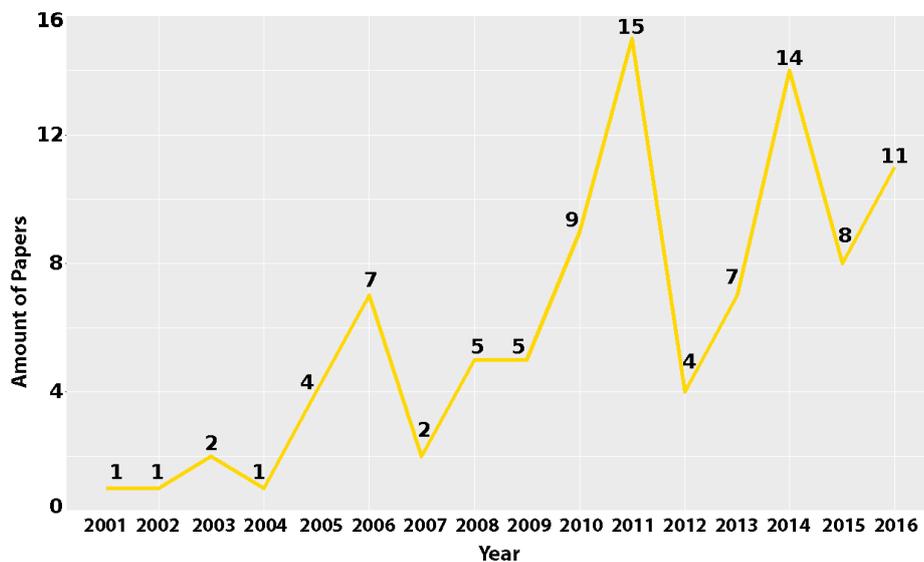


Fig. 3. Distribution of selected papers per year [9].

5.2 Mixed Methods Study

This exploratory study was performed to understand further how iStar extensions are performed and what can be done to help the proposal of future ones. We used mixed methods research [5] since it consists of a qualitative study based on interviews and a quantitative study based on a complementary survey.

The qualitative study performed interviews with 20 participants. Then, the survey was performed with other 30 iStar researchers, different from the ones that participated in the qualitative study, to evaluate the relevance of a set of 18 statements that synthesises the findings of the qualitative study.

The analysis of the qualitative study revealed a good understanding of what extending a modelling language means and pointed out differences about how extensions are proposed. We found out categories that impact positively on iStar extensions (such as reusing existing extensions, proposing extensions in abstract and concrete syntaxes and the creation of modelling tools), and other categories that impact nega-

tively (such as modifying representations of the original constructs, proposing extensions via *ad-hoc* fashion and not carefully choosing graphical representations).

The survey was used to select a subset of the statements, which synthesise the findings of the qualitative study. These selected statements were grouped to generate a set of guidelines to support the proposal of better future iStar extensions.

An interesting result of the qualitative study that should be highlighted is the fact that six participants suggested creating a repository containing the extensions already known. Consequently, our catalogue was proposed motivated by a suggestion of experts of iStar extensions interviewed in this study.

The participants also reported the importance to avoid and mitigating conflicts in the graphical representations of iStar extensions as part of results of the interviews.

5.3 Mitigating Conflicts

We identified 108 conflicts in the concrete syntax of the existing iStar extensions [9]. We classified the conflicts in one of the five categories following:

- **Category 1:** One concept with two or more representations in concrete syntax;
- **Category 2:** Two or more concepts with only one construct in concrete syntax;
- **Category 3:** New Constructs in conflict with the iStar default syntax;
- **Category 4:** Wrong representation of iStar default syntax construct;
- **Category 5:** Representation of constructs that are not part of the extension;

So, motivated by the identification of the conflicts and the point of view of the experts about them, we investigated how to mitigate existing conflicts [11].

For conflicts in category 5 (Graphical representation that is not part of the extension), these representations neither should be considered part of the extension and nor considered on this task of mitigating conflicts. For category 4 conflicts (Wrong graphical representation of iStar default constructs), no further action if needed, we expected future proposers of extensions and tool developers to strictly adhere to concrete syntax provided by iStar 2.0 guide [6].

Category 1 (One concept with two or more graphical representations) comprises redundancy conflicts, which we proposed to be mitigated by surveying potential users to collect preferred representations among the proposals from literature. Overload conflicts of Category 2 (Two or more concepts with only one graphical representation) and Category 3 (New graphical representations in conflict with the iStar default syntax) we proposed to be mitigated by creating new representations and by performing semantic transparency and recognition experiments to identify preferred notations (in a similar study design followed by Caire et al. in [4]).

The results of this study are used to prioritise the representations in our catalogue and to add new representations as an alternative to concepts with two or more representations.

6 CATIE: A Catalogue of iStar Extensions

This section presents the CATIE.

6.1 Extraction of information

Initially, we defined the information fields to be extracted from the iStar extensions based on the analysis of the SLR. These information fields are Id, Title, Link to access the paper, Authors, Type, Name of the Journal/Conference/Book, Extension Base, Application Area, Level of Extension, Compatibility between metamodel and concrete syntax of extensions, Metamodel Completeness, Concepts Definition, Kind of Constructs Proposed, Is there tool support?, Kind of Validation, Reasoning Approach, Static Semantic and Metamodel. Table 1 presents the list of the information to be extracted and the description of each one.

We also established the information of the constructs to be identified in the extensions: Id, Name, Related extension, Description, Objective, Form, Type, Classification, Notation, Example and Conflicts. Table 2 presents the list of the information fields to be extracted and the description of each one.

These fields were used to create spreadsheets to be used in the extraction of the information of the iStar extensions. The information was manually extracted and used to populate the catalogue.

Table 1. The template of the information about the extension.

Information	Description
ID	Code to identify the extension
Title	Title of the paper related to the iStar extension
Link to access the paper	Link to access the paper of the iStar extension
Authors	Authors of the paper related to the iStar extension
Type	Type of the place where the paper was published (Values: Journal/Conference/Book)
Journal/Conference/Book	Name of the place where the paper was published
Extension Base	If the extension is based on other extension, this information links them together.
Application Area	Represents the application area that the extension is proposed to. It can be one or more of the following values (Social, Intelligent, Security, Contextual, Enterprise, General Development, Software Product Lines, Aspects and Other Non-Functional Requirements)
Level of Extension	The level of the representation involved in the proposal of the extension (Abstract syntax, Concrete Syntax or Both)
Compatibility between metamodel and concrete syntax of extensions	To identify if there is compatibility between the representation of new constructs in the metamodel and concrete syntax

Metamodel Completeness	This information is concerned with the absence of nodes and absence of links of the iStar default values in the iStar extension metamodel
Concepts Definition	It specifies if the concepts were totally defined, partially defined or undefined.
Kind of Constructs Proposed	This field analyses if the extension proposed only nodes, only links or links and nodes
Is there tool support?	It represents if the iStar extension created a tool to easier it usage or not
Kind of Validation	The validation presented by the paper. Possible values Illustration, Case study, Experiment, Example of use, No evaluation
Reasoning Approach	Reasoning approaches are commonly used in iStar extensions. Thus, we classified the extensions according to this criterion using the following values: Logic-based, Mathematical formula, Algorithm, Algorithm & Logic-based, No use of reasoning technique
Static Semantic	This is the information if the extension presented static semantic for its metamodel or not
Metamodel	Image of the metamodel (When available in the paper)

Table 2. Template of the information about the constructs.

Information	Description
ID	Code to identify the construct
Name	Name of the concepts which the construct represents
Related extension	The extension that the construct is related to
Priority	Priority is used when there are conflicts of symbol redundancy related to the current construct. Value one presents the greatest priority.
Objective	The purpose of the construct, such as specialise entity.
Description	A textual description of the concept which the construct represents
Form	A name that represents the graphical form of the construct
Type	Node of link
Classification	The classification about the kind of representation. The possible values are a new graphical representation for a node, a new graphical representation for a link or a set of textual representations
Notation	Image with the graphical representation of the construct
Example	A model with the usage of the construct
Conflicts	When the construct conflicts with other constructs, this information is identified here

6.2 Presenting the catalogue

The CATIE is available at <http://istarextensions.cin.ufpe.br/catalogue/>. It groups the iStar extensions and facilitates the search of iStar extensions and their constructs. A text with the presentation of the catalogue is presented initially and the option of analysing Extensions, Constructs, Conflicts and Suggested extensions.

So, the link to the extensions shows the list of all iStar extensions presenting the information of Id, Title, Authors, Year and Source. It is possible to search the iStar extensions by author and title or filter the extensions by application area, reasoning approach used, kind of constructs proposed, kind of validation, the level of extension, tool support, reasoning approach used and well formedness rules definition. It is also possible to search constructs by the name of the construct, application area, the form used, type of construct and objective of representation. It is possible to analyse the iStar extensions in a hierarchical tree view (*Switch to Tree View* button is used to change to this representation) which shows the iStar extensions used as the basis for other ones. Fig. 4 presents the representation of the extensions (Fig. 4 - part A) and tree view (Fig. 4 – part B) and their search and filter fields.

(A) iStar Repository | Extensions | Constructs | Conflicts | Inform Extension | Team Login

Extensions List | Switch to Tree View

Search by Author, Title or Filter by Categories

Author or Title Search and/or Filter By Search

ID	Title	Authors	Year	Source
14	A comparison of goal-oriented approaches to model software product lines variability	Borba, Clarissa; Silva, Carla	2009	International Conference on Conceptual Modeling
72	A Framework for Self-Adaptive Software Based on Extended Tropos Goal Model	Lei, Yiwei; Ben, Kerong; He, Zhiyong	2015	International Conference on Intelligent Human-Machine Systems and Cybernetics

(B) iStar Repository | Extensions | Constructs | Conflicts | Suggest Papers | Team Login

Extension List | Switch to List View

Search

Original I*
Modelling Strategic Relationships to...
Yu, Enc

Considering Role-base...
Chung, Yi C

Security and Trust Req...
Giorgini, Paolo; Massacci, Fabio; Zan...

Tropos: An Agent-Orie...
Bresciani, Paolo; Perini, Anna; Giorgi...

GRANd: A goal-oriente...
Giorgini, Paolo; Rizza, Stefano; Garz...

Fig. 4. Two views of the list of the iStar extensions.

An iStar extension can be selected by clicking on its name in table or tree view to present its details of the information described in Table 1 and the list of the constructs introduced by it. Fig. 1 presents the detailed information of an iStar extension. Each construct can be selected to show details of the extensions by using the option *Detail* (see Fig. 1).

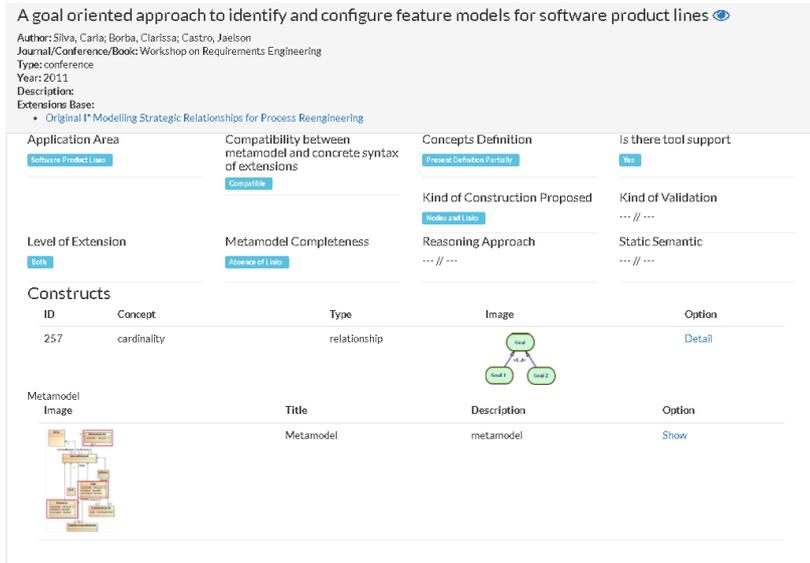


Fig. 1. The information of an iStar extension.

The link to the constructs shows the list of all constructs of the iStar extensions presenting the following information: Application Area, Name of Construct, Form, Type, Notation and Prioritisation. It is possible to search the constructs by title or filter the constructs by Application area, Form, Type and Classification of the kind of representation. A construct can also be selected by the name of the concept (see Fig. 2 – part A) to present its details of the information described in Table 2. Fig. 2 – part B presents the detailing of a construct.

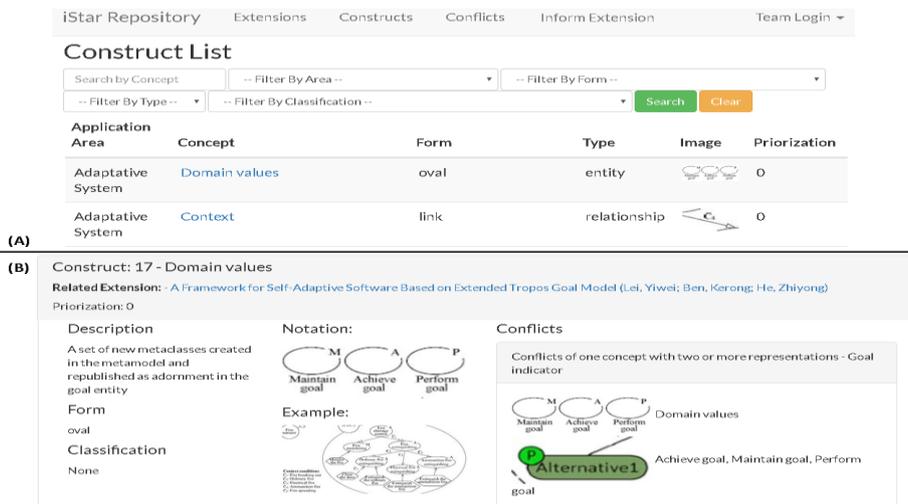


Fig. 2. List of constructs and detailed information of a construct.

The link to the conflicts shows the list of all conflicts of the iStar extensions presenting the information of the description and number of constructs involved. It is possible to filter the conflicts by Type (One concept with two or more representation in concrete syntax, Two or more concepts with only one construct in concrete syntax, New Constructs in conflict with the iStar default syntax, Wrong representation of iStar default syntax or Representation of constructs that is not part of the extension). Fig. 3- part A shows the list of conflicts and the filter field.

A conflict can be selected to present its details of the graphical representations of conflict. The representation of priority of the constructs with conflict and the new representations, which were proposed in [11], can be identified in the information of priority in the constructs and conflicts menus. Fig. 3 – part B presents a conflict in detail.

Conflicts List

Search

Description	Constructs Involved
Conflict of new constructs in conflict with the iStar default syntax	8
Conflict of one concept with two or more representations - Conflicts of Nodes Identifier in Nòmos	4

(A)

(B) Conflict: 1 - Conflict of one concept with two or more representations - Plan
 Type: One concept with two or more representations in concrete syntax

Constructs		Description	Priority	Image
Concept	Form			
Plan	Parallelogram as a arrow	A new representation of this construct based on the results of an experiment.	1	
Plans	octagon	Octagon represents Plans, in the original Tropos is a hexagon	2	
Plans	hexagon	Plans are represented as tasks	3	

Fig. 3. List of conflicts and detailed information of a conflict.

Finally, when a new iStar extension is created, the extender can add it to the catalogue by the option *Inform extension* (Fig. 4 – part A). It is required to inform the title and link to access the extension and the e-mail of the extender. The extensions present in the CATIE have a status associated. The possible values are Submitted, Approved or Rejected. Submitted is the default status when the extension is informed in the catalogue. The status changes to Approved when an expert approves the extension or to Rejected when an expert rejects it. The approval/rejection can be done by the expert using the option team login (top and right side of the Fig. 4). A list of extensions to be approved appears (Fig. 4 – part B) and the expert can approve/reject them (Fig. 4 – part C).

iStar Repository Extensions Constructs Conflicts Inform Extension Team Login ▾

Title of the Extension

Link to Access the Extension

Your E-mail

I'm not a robot  reCAPTCHA
Privacy - Terms

(A)

iStar Repository Extensions Constructs Conflicts Inform Extension Paper Submit Team Dashboard ▾  Enyo José ▾

Show: **Waiting Review**

Title	Year	Author	Journal
Test			

(B)

(C) Test   

Author:
Journal/Conference/Book:
Type:
Year:
Description:

Status: In review

Fig. 4. Inform an iStar extension.

These functionalities can be useful for facilitating the identification of an existing iStar extension or a set of the constructs. Consequently, the CATIE can be used when the extender needs to search this information during the proposal of a new iStar extension. On the other hand, a requirement engineer or a domain expert can use the CATIE to identify a suitable extension to model its systems.

CATIE is part of a process to conduct iStar extensions¹. The catalogue is used to identify if there is an iStar extension which has the concepts to be introduced by a new extension. When an existing iStar extension is identified, the new extension should not be proposed. The catalogue is also used to search and select constructs to be reused and to publicise new iStar extensions and receive the endorsement of the experts.

¹ <http://www.cin.ufpe.br/~ejtg/prise>

7 Conclusion and Future Work

Many extensions have been proposed since the initial iStar version, in the 90's. Consequently, find a specific iStar extension based on its characteristics or find a specific construct are tasks that require extra time and can be unsuccessful. Therefore, it is relevant to analyse these extensions and extract information to easier the identification of them and their constructs. In this paper, we presented the CATIE: a catalogue of iStar extensions. This catalogue is important to facilitate the identification by the extenders of the existing extensions and constructs previously proposed to be used or reused. The CATIE is based on the results of our SLR [9] and the study for mitigation of conflicts [11] which established new representations and prioritisation of the constructs with conflict.

The CATIE is based on the results of an SLR which selected papers until 2016. So, as future work, we intend to perform an update of this SLR considering papers from 2017 and 2018 to update the CATIE. We are working currently in the definition of a process to conduct iStar extensions. This process will use the CATIE to facilitating the find by existing iStar extensions and there is a task to update the catalogue when a new iStar extension will be created. Finally, the constructs of the iStar extensions of the CATIE can be formalised by a conceptual model, metamodel or ontology.

Acknowledgments

The authors thank CNPQ/Brazil (Conselho Nacional de Desenvolvimento Científico e Tecnológico) for the financial support to the execution of this work, Universidade Federal do Ceará (UFC), LER-Universidade Federal de Pernambuco (LER/UFPE) and NOVA LINCS Research Laboratory (Ref. UID/CEC/04516/2013).

References

1. Ali, R., Dalpiaz, F., Giorgini, P Location Based Software Modelling and Analysis: Tropos-Based Approach. International Conference on Conceptual Modelling, Lecture Notes in Computer Science, Volume 5231, pp 169-182 (2008).
2. Amyot, D., Ghanavati, S., Horkoff, J., Mussbacher, G., Peyton, L., Yu, E.: Evaluating goal models within the goal-oriented requirement language. International Journal of Intelligent Systems (2010).
3. Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: Tropos: an agent-oriented software development methodology. Autonomous Agents Multi-Agent Systems Journal (2004).
4. Caire, P., Genon, N., Heymans, P., Moody, D. Visual notation design 2.0: Towards user comprehensible requirements engineering notations. 21st IEEE International Requirements Engineering Conference (2013).
5. Creswell, J. A concise introduction to mixed methods research. Sage Publications (2014).
6. Dalpiaz, F., Franch, X., Horkoff, J.: iStar 2.0 Language Guide. Available in <https://arxiv.org/abs/1605.07767> (2016).

7. Gans, G., Lakemeyer, G., Jarke, M., Vits, T. SNET: A Modelling and Simulation Environment for Agent Networks Based on i* and Congolog. International Conference on Advanced Information Systems Engineering (2006).
8. Giorgini, P., Massacci, F., Zannone, N.: Security and trust requirements engineering. International School on Foundations of Security Analysis and Design III (2005).
9. Gonçalves, E., Castro, J., Araújo, J., Heineck, T.: A Systematic Literature Review of iStar extensions. *The Journal of Systems and Software*, v. 137, p. 1-33, (2018).
10. Gonçalves, E., De Oliveira, M., Monteiro, I., Castro, J., Araújo, J. (Accepted for publication) Understanding what is important in iStar extension proposals: the viewpoint of researchers, *Requirements Engineering Journal* (2018).
11. Gonçalves, E., Almendra, C., Castro, J., Araújo, J., Goulão, M. (Under Review) Using Empirical Studies to Mitigate Conflicts in iStar Extensions, *International Journal on Software and Systems Modeling* (2018).
12. Islam, S., Mouratidis, H., Kalloniatis, C., Hudic, A., Zechner, L. Model Based Process to Support Security and Privacy Requirements Engineering, *International Journal of Secure Software Engineering*, Volume 3, Issue 3 (2012).
13. Moody, D. The “Physics” of notations: towards a scientific basis for constructing visual notations in software engineering. *IEEE Transaction on Software Engineering* 35, N° 5 (2009).
14. Morandini, M., Penserini, A., Perini, A., Marchetto, A. Engineering Requirements for Adaptive Systems. *Requirements Engineering Journal*, Volume 22, Issue 1, pp 77-103 (2015).
15. Palomares, C., Quer, C., Franch, X., Renault, S., & Guerlain, C.: A catalogue of functional software requirement patterns for the domain of content management systems. 28th ACM Symposium on Applied Computing (2013).
16. Peixoto, M., Silva, C. A Gamification Requirements Catalog for Educational Software: Results from a Systematic Literature Review and a Survey with Experts. 32nd ACM Symposium on Applied Computing (2017).
17. Siena, A., Maiden, N., Lockerbie, J., Karlsen, K., Perini, A., Susi, A. Exploring the Effectiveness of Normative i* Modelling: Results from a Case Study on Food Chain Traceability. International Conference on Advanced Information Systems Engineering (2008).
18. Toval, A., Olmos, A., Piattini, M. Legal Requirements Reuse: A Critical Success Factor for Requirements Quality and Personal Data Protection, *IEEE Joint International Conference on Requirements Engineering*, pp. 95-103 (2002).
19. Yu, E.: *Modelling Strategic Relationships for Process Reengineering*. University of Toronto, Toronto (1995).