UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

# PATTERN-BASED ONTOLOGY REVISION

Wander dos Santos Vasconcellos

**Orientadores**

Kate Cerqueira Revoredo

Fernanda Araújo Baião

RIO DE JANEIRO, RJ - BRASIL
Setembro de 2014

PATTERN-BASED ONTOLOGY REVISION

Wander dos Santos Vasconcellos

DISSERTAÇÃO APRESENTADA COMO REQUISITO PARCIAL PARA OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO (UNIRIO). APROVADA PELA COMISSÃO EXAMINADORA ABAIXO ASSINADA.

Aprovado por:

_______________________________

Kate Cerqueira Revoredo, D.Sc. - UNIRIO

_______________________________

Fernanda Araújo Baião, D.Sc. - UNIRIO

_______________________________

Flávia Maria Santoro, D.Sc. - UNIRIO

_______________________________

Aline Marins Paes Carvalho, D.Sc. - UFF

RIO DE JANEIRO, RJ - BRASIL
Setembro de 2014

Vasconcellos, Wander dos Santos.

V331 Pattern-Based Ontology Revision / Wander dos Santos Vasconcellos, 2014.

xiii, 122f.; 30 cm

Orientadora: Kate Cerqueira Revoredo.

Coorientadora: Fernanda Araújo Baião.

Dissertação (Mestrado em Informática) - Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2014.

1. Ontologia (informática). 2. Evolução de ontologia. 3. Especificação - Projeto auxiliado por computador. 4. Revisão de teoria. I. Revoredo, Kate. II. Baião, Fernanda Araújo. III. Universidade Federal do Estado do Rio de Janeiro. Centro de Ciências Exatas e Tecnologia. Curso de Mestrado em Informática. IV. Título.

CDD - 006.312

À minha Família e meus Amigos.

# Agradecimentos

Agradeço as minhas orientadoras, Kate e Fernanda, pela paciência e dedicação que me ajudaram a chegar até aqui. Aos amigos do mestrado e do trabalho pelas sugestões e explicações que me ajudaram a superar minhas deficiências. Aos meus pais pelo apoio a minha decisão de voltar a estudar, mesmo sabendo dos sacrifícios que isto representaria. E a minha esposa e ao meu filho por lembrarem-me que minha vida não estava resumida apenas ao estudo.

Vasconcellos, Wander dos Santos. **Pattern-Based Ontology Revision**. UNIRIO, 2014. 122 páginas. Dissertação de Mestrado. Departamento de Informática Aplicada, UNIRIO.

## RESUMO

Ontologia representa um domínio de conhecimento por meio de conceitos, relacionamentos e axiomas. A fim de obter ontologias com boa qualidade, padrões de projeto em ontologias (ODPs) podem ser considerados, originando ontologias baseadas em padrões. Com o passar do tempo, os conceitos e relacionamentos do domínio do discurso evoluem, assim a ontologia correspondente tem que evoluir também. Geralmente, quando esta evolução é realizada manualmente ou semiautomaticamente, ela é sem planejamento, demorada e sujeita a erros. Portanto, mecanismos automáticos são necessários para melhorar os processos de evolução de ontologias. Na área de Programação Lógica Indutiva (ILP), técnicas de Revisão de Teoria são usadas para evoluir automaticamente um conjunto de cláusulas em Lógica de Primeira-Ordem (FOL), através do uso de exemplos para direcionar onde nas cláusulas uma modificação tem que ser feita. Nesta dissertação, duas propostas são apresentadas: (1) revisar automaticamente ontologias baseadas em padrões, pela aplicação de técnicas de Revisão de Teoria e (2) usar os padrões de projeto em ontologia para reduzir o espaço de modificações possíveis. Os resultados obtidos pela aplicação destas propostas foram avaliados através de um experimento, que identificou e analisou os impactos.

**Palavras-chave:** Ontologia; Evolução de Ontologia; Padrões de Projeto em Ontologias; Revisão de Teoria.

**ABSTRACT**

Ontology represents a domain of knowledge, through concepts, relationships and axioms. In order to achieve ontologies with good quality, ontology design patterns (ODPs) can be considered, thus generating pattern-based ontologies. Over time, concepts and relationships of the domain of discourse evolve, thus the corresponding ontology must evolve as well. Usually, when this evolution is performed manually or semiautomatically, it is without planning, time-consuming and error-prone. Therefore, automatic mechanisms are required to improve ontology evolution processes. In the Inductive Logic Programming (ILP) area, techniques of Theory Revision are used to automatically evolve a set of first-order logic (FOL) clauses through the use of examples to guide where in the clause a modification must be done (revision points). In this dissertation, two proposals are presented: (1) automatically revise pattern-based ontologies, by applying Theory Revision techniques and (2) use ontology design patterns to reduce the space of possible modifications. The results obtained by applying these proposals were evaluated by an experiment, which identified and analyzed the impacts.

**Keywords:** Ontology; Ontology Evolution; Ontology Design Patterns; Theory Revision.

# Table of Contents

# List of Tables

# List of Figures

# Glossary

| | |
|---|---|
| DL | Description Logic |
| FCA | Formal Context Analysis |
| FDT | Fundamental Domain Theory |
| FOL | First-Order Logic |
| FORTE | First-Order Revision Theory from Examples |
| GUC | Generic User Case |
| ILP | Inductive Logic Programming |
| ODPs | Ontology Design Patterns |
| OWL | Ontology Web Language |
| OWL-DL | Ontology Web Language - Description Logic |
| RAA | Role Assertion Analysis |
| THY | Theory |

# 1. Introduction

Ontology is an explicit specification of a conceptualization, which is a simplified abstraction of the modeled domain knowledge [1]. Ontology is considered formal when the conceptualization is developed axiomatically, and systematically, in this case the ontology is also called a logical theory [2]. A domain ontology is an artifact which encodes knowledge about a domain, it describes the relevant domain entities and their relationships through the definition of concepts and relationships, respectively. Knowledge-based systems are softwares able to find implicit implications from the knowledge explicitly represented [3]. The Ontologies play a key role in knowledge-based systems [4], because they describe the domains and provide important inputs, which when considered allow more precise inference. So there is an increasing demand especially for large, complex, expressive and high-quality ontologies [5]. An ontology has high quality if it meets the following criteria [1]: clarity, consistent, extensibility, minimal dependence code and minimal ontological commitment. One way to get a high quality ontology is using the ontology design patterns (ODPs) [6] that consist on a set of solutions that solve recurrent ontology modeling problems [6, 7, 8]. An ontology designed using one ODP or more is called pattern-based ontology [4], then it is a high quality ontology.

Ontology engineering is the area dedicated to the research of the principles, methods and tools for developing and maintaining ontologies. As part of knowledge-based systems, the ontologies, like any software artifact, has a life cycle of its own. This life cycle [9] consists of five phases: feasibility study; kickoff; refinement; evaluation; application and evolution. Ontology evolution is used to correct an ontology that requires a combination of domain experts, ontology engineers, and other resources. Its goal is to transform an ontology that has become incorrect or incomplete due changes on the context. An ontology is considered incomplete with respect to a subset of data that should be covered, if not all concepts and relationships needed to model these data are present, and incorrect when an inference, performed by the ontology, is wrong. As an example consider the Family

Law Domain. According to the Brazilian Civil Code, civil marriage can only be realized between persons of different genders, but this has changed when the Brazilian Supreme Court ruled that same-sex couples can marry civilly. So, an ontology that represents this domain must go through an evolution to correctly represent the domain, according to the decision of the Brazilian Supreme Court. Although in this example the required change seems simple, in reality changes in a context may require extensive and complex modifications in several points of the ontology not to mention that this can become recursive, turning it into a complex task.

Automatic [10, 11, 12, 13, 14, 15] and semiautomatic [15, 16, 17, 18, 19] mechanisms for ontology evolution have been proposed in the literature. These mechanisms, in different ways, propose to evolve an existing ontology with the goal of making it to represent the domain knowledge correctly again. The evolution process is handled differently by these mechanisms. Some ignore the ontology and perform the evolution from scratch [11, 12], others consider the whole ontology and look for failures on it [13, 14, 15, 16, 17, 18, 19]. Finally, some consider it almost correct and complete and only to carry out minor changes [10].

Semiautomatic or manual ontology evolution may be time-consuming, costly and error-prone [20]. On the other hand, theory revision is the task oriented to automatically transform an incomplete or incorrect set of first-order clauses (theory), in a complete and correct theory using a set of examples [21]. It is a process that minimally modifies a theory aiming at making it consistent with a dataset composed of positive and negative examples. Positive examples are facts that must be proved by theory and negative examples are incorrect facts that must be refuted. Therefore, the dataset of examples is used to identify and correct the failure points of the theory.

During the preparation of this dissertation, some papers were produced according to the development of the work. In [22], an overview of the proposal approach was presented and in [23] the proposal was detailed and a preliminary experiment presented.

## 1.1 Motivation and Relevance

The ontology evolution is a fundamental stage in the life cycle of an ontology, because it ensures that the ontology correctly represents the domain knowledge after a change in the ontology context. This task is more difficult when the ontology is large, complex, capable of performing inferences and has high quality, because the large number of concepts, relationships and axioms represented, requires a greater effort of evolution. So, reduce this effort of evolution is important for increasing the efficiency of the process.

On the other hand, the ODPs can be viewed as a way to express the high quality of an ontology, because they are solutions to recurrent ontology modeling problems. So, it is important that ODPs used in a pattern-based ontology be preserved during the process of evolution so that the ontology's efficiency and understandability are maintained.

## 1.2 Problem

The changes of context can cause the need for evolution of ontologies, because it is important that the inference remain accurate, for example. Mechanisms of ontology evolution that either address their task from scratch or manipulate the whole ontology to discover and correct errors may not be the most appropriate way to evolve a large, complex and high quality ontology when it is almost correct and complete. When the process of ontology evolution is performed on a large and complex pattern-based ontology almost correct and complete is important that its ODPs be preserved after the evolution of ontology, for reasons of efficiency and understandability, without preventing the ontology is corrected.

## 1.3 Hypothesis

Considering the motivation and relevance (section 1.1) and the problem (section 1.2) of how to evolve an pattern-based ontology aiming to correctly describe the domain of discourse, the goal of this dissertation is to preserve the ODPs used in an pattern-based ontology during a process of evolution to maintain and preserve the high quality of them. To achieve this goal, it was developed a proposal for automatic ontology evolution that attempts to preserve the good quality of the ontology, while maintaining it consistent with the knowledge domain.

Therefore, this dissertation should confirm the following hypothesis:

***If*** *the process of pattern-based ontology evolution mantains the occurrences of ODPs,* ***then*** *the revision space will be reduced and a complete and correct pattern-based ontology will be obtained.*

## 1.4 Proposal

The proposal of this dissertation consists of: a method of conversion of ODPs to first-order logic (FOL) and a process for pattern-based ontology evolution, with a subjacent supporting tool. The method of conversion allows that ODPs of an pattern-based ontology are automatically identified. The process of pattern-based ontology evolution consists of two stages: pre-revision and revision. In the first stage, the ODPs are automatically identified and selected for protection. This selection can be performed automatically, when all identified ODPs are protected or semi-automatically where an expert selects the ODPs to be protected. The second stage performs the ontology revision itself through a theory revision system.

In this dissertation, the Pre-REvision THeorY (PRETHY), a proposal for performing automatic pattern-based ontology revision able to maintain the good quality of an ontology, through the protection of ontology design patterns (ODPs) is proposed. The PRETHY is based on a repository of ODPs, modeled in first-order logic (FOL), used in the pre-revision stage that automatically identifies the ODPs instantiated in theory to protect them from being modified during the theory revision stage.

## 1.5 Scientific Method

The scientific method applied to prove the hypothesis of this dissertation is the experiment, which gives it an explanatory nature. It will be a quantitative research, where data will be collected in several experiments as follows: 1) experiment on the proposal of this dissertation, with a quantitative and explanatory characteristic; 2) quantitative comparison with another process of ontology evolution. These data will then be analyzed and explained, so that the confirmation of the hypothesis is drawn.

## 1.6 Organization of the Dissertation

Besides this introduction, this dissertation consists of the following chapters. In chapter 2, basic concepts for understanding the proposal are presented, while in chapter 3 the proposal itself is detailed. In chapter 4, the experiment and its results are presented. In chapter 5, related works to this dissertation are presented and analyzed and in chapter 6, contains the conclusion of this dissertation and suggestions of future works.

# 2. Theoretical Foundation

For a better understanding of the proposal of this dissertation some concepts are important, such as ontology [1], ontology design patterns (ODPs) [6] and theory revision [24].

## 2.1 Ontology

A domain ontology is an artifact which encodes knowledge about a domain. It describes the relevant domain entities and their relationships through the definition of concepts and relationships, respectively [1]. For example, in Figure 2.1 an ontology for the travel domain is depicted. It forms a graph where nodes represent concepts (Thing, Accommodation, AccommodatioRating, Activity, Contact, Destination, BedAndBreakfast, Campground and Hotel) and edges represent the relationships (hasPart, hasAccommodation, hasActivity, hasContact, hasRating and isOfferedAt).

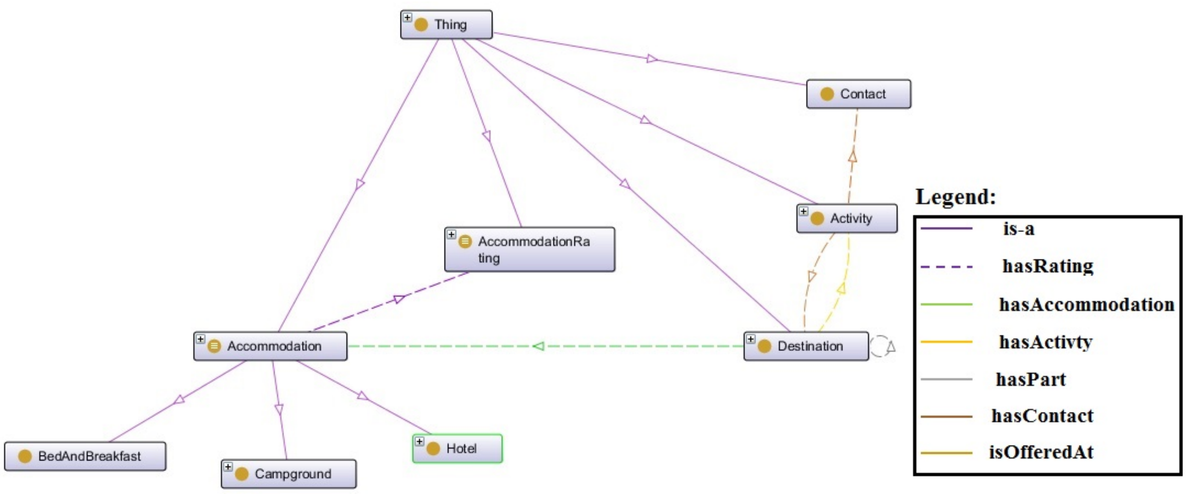


Figure 2.1: Example of ontology (obtained from protégé ontology repository)

Relationships represent the interactions between concepts, concepts' properties or concepts' instances. There are two types of relationships: taxonomic and non-taxonomic. Taxonomic relationships define a hierarchy among the modeled concepts, thus defining concepts and sub-concepts structure. In the travel domain (Figure 2.1), examples of taxonomic relationships are all the violet edges, except the violet dotted edge (hasRating). For example, a Hotel is a sub-concept of the concept Accommodation . Non-taxonomic relationships are those that do not define a hierarchy between concepts. In the travel domain a destination can be composed of destinations (self-relationship hasPart, gray edge), a destination has accommodations (hasAccommodation, green edge), and destination also has activities (hasActivity, orange edge). Accommodations have ratings (hasRating, violet dotted edge), an activity can be offered to many destinations (isOfferedAt, light brown edge) and an activity can have many contacts (hasContact, dark brown edge).

Ontologies, like any software artifact, has a life cycle of its own. This life cycle [9], consists of:

**Feasibility study** - In this phase, factors outside the area of information technology, which may determine the success of a knowledge management application are analyzed, for example, to identify problems, opportunities and potential solutions. It serves as a decision support economical and technical, determining the area focus and target solution.

**Kickoff** - In this phase, the ontology requirements specification documents (ORSD) and a semi-formal description of the ontology are prepared. Knowledge sources (eg, text documents, relational databases, etc.) are also gathered for using in the generation of semi-formal description of the ontology and the refinement phase. The ORSD describe what the ontology should support and guide the ontology engineer in the decision to include or exclude concepts, relationships and the hierarchical structure of the ontology. It is also in this phase that it should be determined the reuse of already developed ontologies.

**Refinement** - In this phase, the semi-formal description of the ontology is formalized from the knowledge sources, generating the target ontology. The two most common approaches to perform the refinement are the top-down and the bottom-up strategies. In the top-down approach, concepts and relationships are modeled in a very general level, for subsequent refinements and this approach is generally performed manually. In the bottom-up approach, concepts and relationships are modeled in a very detailed level, is generally applied with the support of automatic or semiautomatic tools for ontology learning, for example, Text-2-Onto [12] and LexO [25].

**Evaluation** - This phase can occur at different times in the life cycle of ontology: refinement, evolution and application. It addresses three aspects: technology, user and model. The first evaluates issues such as interoperability, scalability language conformity, etc. The second evaluates whether the users are satisfied with the application. And the third seeks to avoid modeling errors.

**Application and Evolution** - the application sub-phase is done through the use of the ontology based systems and comprises the following steps: creation/import of knowledge, capture of knowledge items (to define the importance and linkages of knowledge) and retrieval and access to knowledge. The evolution sub-phase is an process used to correct an ontology that requires a combination of domain experts, ontology engineers, and other resources. Ontology engineers before deploying a new version of an ontology must test it thoroughly to know all the possible effects of the changes made. When the ontology evolution is performed manually is important to define its frequency and how it will be performed. Automatic ontology evolution approaches for consistence maintenance of ontologies can also be adopted.

The ontology engineering process seeks reusing existing ontologies to increase its efficiency and effectiveness, and can also contribute to good quality of the ontology. For Gruber [1], an ontology has high quality if it meets the following criteria:

**Clarity** - Informs the intended meaning of defined terms. The definition should be independent of the social and computational contexts, formal and complete.

**Consistent** - The ontology should be consistent. This means that the inferences made by it are coherent with their definitions. For this, its axioms must be logically correct.

**Extensibility** - Allows one to define new terms without needing to review the existing terms.

**Minimal dependence coding** - The conceptualization should not depend on coding symbols.

**Minimal ontological commitment** - Are agreements to use the vocabulary of the ontology of shared form.

## 2.2 Ontology Design Patterns

Ontology design patterns are modeling solutions with recognized good quality that solve recurrent ontology modeling problems, such as lack of expressivity of the logical formalism [6, 7, 8] that may be applied to improve the quality of an ontology[6]. To illustrate this, consider the ontology shown in Figure 2.2:

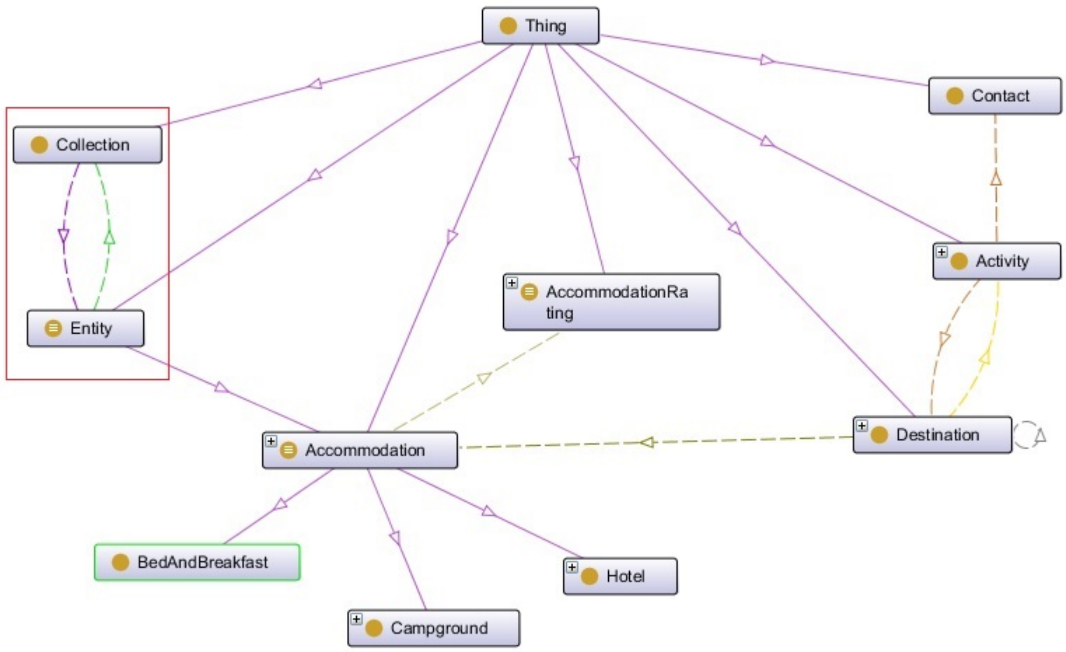


Figure 2.2: Ontology with ODP

Comparing the ontology of Figure 2.2 with the ontology of Figure 2.1, which represent the same domain, what differentiates them is the existence of an ODP at the ontology in Figure 2.2. The pattern Collection consists of the Entity and Collection concepts. It models collections of entities and with this pattern, you can see that the accommodations are entities and form one or more collections in the domain shown in Figure 2.2. For example, accommodations could be grouped into collections of expensive and cheap accommodations. But the most important is related to the gain in quality and expressiveness of an ontology through the use of an ODP, because it is more clear to the user that the accommodations form collections and inferences can be made in this direction.

According to section 2.1 ontology engineering seeks to benefit from reuse of existing ontologies to ensure the efficiency and effectiveness of the process, besides the good quality of the produced ontology. One way to ensure these benefits is through the reuse of ODPs that are good quality solutions to recurring problems of modeling. An ontology produced through the reuse of ODPs is called pattern-based ontology [4].

The ODPs are a set of good practices that can be used for ontology construction, maintenance, interoperability, understanding by the user, among other, for the purpose of getting a product of good quality. Their concepts are based largely on the work on design patterns in software engineering [26] and can be grouped into six groups [7]:

1. structural;
2. correspondence;
3. reasoning;
4. presentation;
5. lexical-syntactic;
6. content.

The structural patterns are focused on expressiveness and are divided into: logical and architectural. Logical patterns are ways of solving problems of expressiveness of the logical formalism used. And architectural patterns affect the general format of an ontology, restricting it and the logical constructors used. Examples of architectural patterns are: taxonomy and modular architecture [7].

Correspondence patterns are focused on interoperability and are divided into: reengineering and mapping. The former are focused on providing solutions for the processing of non-ontological conceptual models (for example, UML diagrams) in new ontologies. And the latter are used to create semantic associations between two existing ontologies [7].

Reasoning patterns are driven by results. When declared previously, they allow the systems to decide what kind of reasoning should be performed to obtain the expected results. Examples of reasoning patterns are: classification, subsumption, inheritance, materialization, among others [7].

Presentation patterns focus on usability and readability of ontologies from the user's perspective. They are considered good practice to support reuse of patterns for facilitating the evaluation and selection of them. These patterns are divided into: naming and annotation. The first defines conventions for creating names of the elements in an ontology. The last provides annotation properties used to improve the comprehensibility of ontologies and their elements. Labels and comments are types of this pattern [7].

The Lexical-syntactic patterns are linguistic structures composed of words arranged in a specific order, which allows us to generalize and draw some conclusions about what they express [7].

Content patterns focus on specific problems involving concepts and relationships pertaining to domain [6]. They are small pieces of ontologies or small ontologies, language-independent, that can be reused in other domains. Upper and core ontologies are the origins of these patterns [8, 26]. Upper ontologies are domain independent, describe generic concepts and can be reused to construct new ontologies. A core ontology already provides a right definition of a structural knowledge, of a specific field, used in various domains of application [26]. The proposal of this dissertation is mainly focused on ensuring the consistency criterion (see section 2.1), with the aid of content ODPs.

Content patterns are defined from the generic use cases (GUCs) [7]. A GUC should be generic enough to allow the representation of models of various problems, which means it can be used in many real cases [27]. A knowledge domain can contain several use cases and a use case can participate in various knowledge domains [7]. Use cases are formed by competency questions that are questions that an expert can make to the ontology and that it has to answer. Therefore, a correct ontology should specify only the concepts and relationships needed to answer all competency questions laid down for their use cases [7, 27].

Among the content patterns, that are available in the ODP public repository[1] on the Internet, there are: Participation, Co-Participation, Communities and Collection.

Participation is the pattern that represents the participation of an object in an event. Using cardinality constraints, this ODP enables the definition of the number of participants in an event, the events in which an object participates in, among other pieces of knowledge. An object is some physical, social or mental object or substance. And an event is some physical, social or mental process, event or state. For example, a surgeon participates in a surgery. This pattern has the following competency questions to answer: Which objects do participate in this event? and Which events do this object participate in?. The Figure 2.3 is the diagram of this pattern.

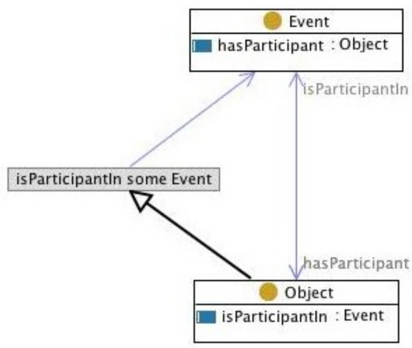


Figure 2.3: Participation Content ODP

Co-Participation is the pattern that represents the participation of two objects in the same event. This pattern is based on Participation pattern. An object is some physical, social or mental object or substance. And an event is some physical, social or mental process, event or state. For example, the participation of the pilot and co-pilot on a flight.

[1] www.ontologydesignpatterns.org

This pattern has the following competency questions to answer: What objects participate in a same event? and Who is involved with whom in something?. The Figure 2.4 is the diagram of this pattern.

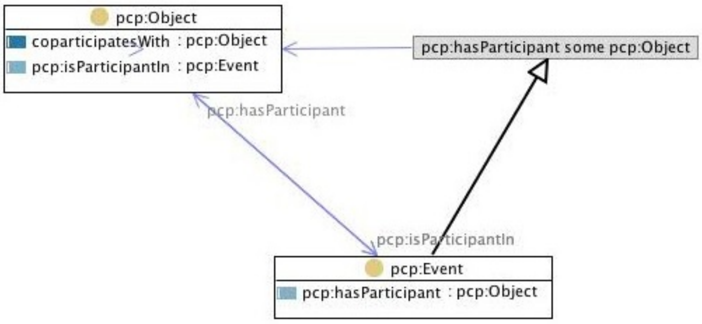


Figure 2.4: Co-participation Content ODP

Communities is the modeling pattern of a community whose members are agents. An agent is some executor object of action, whether physical or social. For example, a community of nurses. Further, a community may be an element of another community.

Collection is the modeling pattern that represents collections and their entities (members). A collection represents the concept of a set of entities (things) that share one or more properties. An entity is anything real, possible or imaginary to model. For example, a collection of books. This pattern has the following competency questions to answer: What things are contained in this collection (community collective)? and What collections this thing is member of?. The Figure 2.5 is the diagram of this pattern.

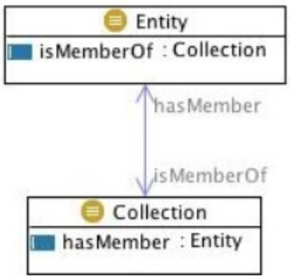


Figure 2.5: Collection Content ODP

## 2.3 Theory Revision

The acquisition of knowledge is a difficult task, time consuming and error prone. The process of improving automatically a knowledge base using learning methods can be achieved through theory refinement. Theory refinement consists of two tasks [21]:

theory restructuring and theory revision. Theory restructuring is the task oriented to the improvement of performance or understandability by the user of the theory. The theory revision is the task oriented to transform incomplete or incorrect theories in a coherent theory from a set of examples. For more details about theory restructuring, see [28]. In this dissertation, will be covered the task of theory revision whose goal is:

- Given a knowledge base composed by a set of first-order Horn clauses[2] (theory), which can be incomplete or incorrect.
- Modify minimally this theory aiming at make it consistent with a dataset of examples.

A theory is considered incomplete if not all concepts and relationships needed to model the domain are present and incorrect when an inference, performed by the theory, is wrong. The dataset is composed of positive and negative examples, usually represented by ground facts. The former are instances of objects or relationships deemed correct and, therefore, must be confirmed by the theory, while the latter are considered incorrect instances and, therefore, must be refuted by the theory. A theory revision system starts from the evaluation of an initial theory, where this can be divided in two parts: background knowledge (henceforth called FDT), which is assumed to be correct, and another part that can be modified by the revision (henceforth called THY). An example of theory revision system is FORTE (First-Order Revision Theory from Examples) [24, 29]. The top-level algorithm is depicted in Algorithm 1.

[2] A first-order Horn clause is a clause (disjunction of literals) with at most one positive literal.

**Require:** Initial Theory composed by FDT and THY ; Dataset
**Ensure:** Final Theory composed by FDT and a revised THY (THY')

**repeat**
  generate revision points;
  sort revision points by potential (high to low);
  **for all** revision point **do**
    **if** the potential of current revision point is less than the score of the best revision to date **then**
      break;
    **end if**
    generate revisions;
    update best revision found;
  **end for**
  **if** best revision improves the theory **then**
    implement best revision;
  **end if**
**until** no revision improves the theory

**Algorithm 1:** FORTE's top-level refinement algorithm (adapted from Richards and Mooney [24])

The adaptation made in the original algorithm ensures that only revision points that can improve the accuracy of the theory in current state will be evaluated. The first step is the generation of revision points. These are generated when the initial theory is tested by a set of examples. Each revision point has its potential calculated, that is the number of examples misclassified. Thus, the set of revision points is decreasingly ordered by the potential. The set of revisions is then generated from the ordered set of revision points. Each revision receives a score that is the actual increase in theory's accuracy that it achieves. The best revision that is one that most improves accuracy, is selected. If it improves the theory's accuracy, it is implemented. If no improvement of the theory's accuracy is obtained, the cycle stops, otherwise the cycle begins again.

The revision points are points in the theory where a revision has the potential to improve the theory's accuracy. They can be of two types: *generalization* and *specialization* revision points. The first represents failures in the evaluation of positive examples and the second represents failures in the evaluation of negative examples. In order to correct these failures, *revision operators* are used. Generalization revision points are corrected through generalization revision operators, such as add-clauses, delete-antecedents, identification and absorption. The goal of generalization operators is to allow all positive examples be proved. Specialization revision points are corrected through specialization revision operators, such as add-antecedents and delete-clauses. The goal of specialization operators is to ensure that all negative examples accepted are rejected [21].

Consider the following logical theory from the travel domain described in section 2.1:

1. hasActivity(Dest, Act) :- isRelaxation(Dest, Act).
2. hasActivity(Dest, Act) :- isAdventure(Dest,Act).
3. hasActivity(Dest, Act) :- isSightseeing(Dest, Act).
4. hasActivity(Dest, Act) :- isSports(Dest, Act).
5. isBeach(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), isRelaxation(Dest, Act), hasContact(Act, Con).
6. isBackpackers(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), hasActivity(Dest, Act).
7. isBackpackers(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), hasActivity(Dest, Act), hasContact(Act, Con).

The represented domain portion refers to the concept of destination (Destination) and their relationships (isBeach, isBackpackers, hasAccommodation, hasActivity). Now, consider a scenario where when the destination is a beach, activity is relaxing and there are not contacts. In this case, clause (5) is wrong and positive examples as **isBeach(dest1,acc1,act1)** are not proved and a revision point (generalization) will be generated for the antecedent (**hasContact(Act, Con)**).

To correct this revision point, generalization operators must be used, such as delete-antecedent, add-rule, identification or absorption. The delete-antecedent operator is divided into two types: hill-climbing delete-antecedent and delete multiple antecedents. The first excludes one antecedent at a time and checks if the deleted antecedent allows to prove more positive examples while does not allow to prove any negative example. The second excludes many antecedents simultaneously. For this, it selects only those antecedents whose individual exclusion does not increase the proof of positive examples, but does not increase the proof of negative examples [24]. For example, the clause (5) generated a generalization point. This point is solved by delete-antecedent operator applied to it and would produce the following solution:

- isBeach(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), isRelaxation(Dest, Act).

The add-rule operator generates new versions of an existing rule without changing the original rule. For example, still considering that clause (5) is wrong, the following solution is obtained by this operator:

- isBeach(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), isRelaxation(Dest, Act).
- isBeach(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), isRelaxation(Dest, Act), hasContact(Act, Con).

The identification operator creates a new rule to an existing predicate. Its goal is to generalize an antecedent that caused the failure of the proof of a positive example [24]. To illustrate the use of the identification operator, consider the following clauses:

1. calmDestination(Dest, Act) :- hasContact(Act, Con), isCalmActivity(Act, Dest).
2. calmDestination(Dest, Act) :- hascontact(Act, Con), yogaActivity(Act, Dest).
3. isCalmActivity(Act, Dest) : sunBathingActivity(Act, Dest).

When applying the identification operator, the second clause is deleted and a new rule for isCalmActivity predicate is added, yielding the following solution:

1. calmDestination(Dest, Act) :- hasContact(Act, Con), isCalmActivity(Act, Dest).
2. isCalmActivity(Act, Dest) :- yogaActivity(Act, Dest).
3. isCalmActivity(Act, Dest) :- sunBathingActivity(Act, Dest).

And to illustrate the use of the absorption operator, suppose that a contact or is made exclusively electronically (email) or exclusively by conventional address. So, consider the following clauses:

1. hasContact(Act, Con) :- activity(Act), hasZipCode(Con, Zipc), hasEMail(Con, EMa), isContactOf(Con, Act).
2. contact(Con) :- hasCity(Con, Cit), hasStreet(Con, Stre), hasZipCode(Con, Zipc).
3. contact(Con) :- hasEMail(ConEMa).

No positive example will be proved by clause (1) defined above. In applying the absorption operator, it realizes that the clauses (2) and (3) can help in correction of clause (1). So, the operator replaces the first clause by the following:

1. hasContact(Act, Con) :- activity(Act), contact(Con), isContactOf(Con, Act).

In a scenario where a backpacker destination, activities must have contacts, the clause (6) is wrong because it does not require the presence of a contact. Negative examples will be proved as **isBackpackers(dest1,acc1,act1)** and the entire clause (6) will be a revision point (specialization), because it is too generic.

To correct this revision point, specialization operators must be used, such as delete-rule or add-antecedent. The delete-rule operator is the simplest form of specialization. For example, the clause (6) would be deleted by this operator to solve the revision point.

The add-antecedent operator is the most complex form of specialization. During this specialization process, the addition of an antecedent may cause the rejection of positive examples. In these cases, the add-rule operator is called to generate new clauses from the original clause to prove the other positive examples [24]. For example, consider now that the backpacker destination must have a maximum rating of one star. To solve the specialization point generated by clauses (6) and (7) with the add-antecedent operator, the following solution is obtained:

- isBackpackersDestination(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), hasActivity(Dest, Act), hasRating(Acc, oneStar).
- isBackpackersDestination(Dest, Acc, Act) :- hasAccommodation(Dest, Acc), hasActivity(Dest, Act), hasContact(Act, Con), hasRating(Acc, oneStar).

# 3. Pattern-Based Ontology Revision

In this chapter, an approach for pattern-based ontology revision is presented. For elaboration of this dissertation, the ontologies are expressed in first-order logic. Nowadays, the ontologies are usually built using OWL (Ontology Web Language), description logic, which are a subset of first-order logic [3]. So, this consideration can be made, because it is possible to completely convert an ontology in description logic for a logical theory in first-order logic. Thus, in the section 3.1, the methodology created to describe the ODPs in FOL is presented. Following the proposal for ontology revision is presented in section 3.2. This work does not address the conversion of ontologies for FOL as in [10].

## 3.1 Modeling Ontology Design Patterns in First-Order Logic

In this section, the proposal to convert to FOL any ODP modeled in another formalism is presented. It is important that the conversion process be standardized, because it allows content ODPs to be automatically identified in the ontology. The method used to model content ODPs in FOL consists of 4 steps:

1. Define unary predicates from concepts.
2. Define n-ary predicates (relations) either from existing relationships between concepts or from existing roles that concepts play in relationships.
3. Define n-ary predicates (relations) from competency questions.
4. Make connections between the elements of the patterns with the concepts and relationships of the domain by defining the rules (clauses) required.

The main goal of this method is to identify the concepts and relationships pertaining to patterns and link them to the concepts and relationships of the domain by defining rules. For example, consider the co-participation ODP that will be used in the experiments presented in chapter 4. It has two concepts (see Figure 2.4): object and event. This pattern also has one relationship with two roles: isParticipantIn and hasParticipant.

According to the step (1) of the methodology, concepts are transformed into unary predicates. Then:

- object(O).
- event(E).

According to the step (2) n-ary predicates are defined either by the relationships or by roles played by the concepts in relationships. So, in this ODP, are used the two roles that define the relationship between the object and event concepts: hasParticipant and isParticipant. Therefore, this roles will generate two n-ary predicates as shown below:

- isParticipantIn(O, E).
- hasParticipant(E, O).

As described in section 2.2, the content standards are formed by use cases and these by competency questions that the ontology should be able to answer. The content ODP has a question of competence not identified in its figure: What objects participate in the same event ? This question was modeled according to step (3), generating the following n-ary predicate:

- coParticipatesWith(O1, O2, E).

The step (4), the rules that form the pattern are generated. See in Figure 3.1 the complete definition of three ODPs in FOL.

```
Co-participation First-order modeling:
event(E) :- ?(E).
object(O) :- ?(O).
isParticipantIn(O, E) : event(E), object(O), ?(O, E).
hasParticipant(E, O) :- isParticipantIn(O, E).
coParticipatesWith(O1, O2, E) :- isParticipantIn(O1, E), isParticipantIn(O2, E).
Communities First-order modeling:
member(M) :- ?(M).
community(C) :- ?(C).
isMemberOf(M, C) :- community(C), member(M), ?(M, C).
hasMember(C, M) :- isMemberOf(M, C).
Participation First-order modeling:
object(O) :- ?(O).
event(E) :- ?(E).
isParticipantIn(O, E) :- event(E), object(O), ?(O, E).
hasParticipant(E, O) :- isParticipantIn(O, E).
```

Figure 3.1: Final result of the proposed modeling method.

The question marks should be replaced by predicates referring to the concepts and relationships of the modeled domain. For example, in a domain where a *person* is an object, a *matrimony* an event, the relationship between person and matrimony described by the *participatesIn* fact, a *magistrate* is a member, a *group* is a community and the relationship between magistrate and group described by the *belongs* fact, the participation, co-participation and communities ODPs would be instantiated as follows in Figure 3.2:

```
Co-participation instantiated:

event(E) :- matrimony(E).
object(O) :- person(O).
isParticipantIn(O, E) : event(E), object(O), participatesIn(O, E).
hasParticipant(E, O) :- isParticipantIn(O, E).
coParticipatesWith(O1, O2, E) :- isParticipantIn(O1, E), isParticipantIn(O2, E).

Participation instantiated:

object(O) :- person(O).
event(E) :- matrimony(E).
isParticipantIn(O, E) :- event(E), object(O), participatesIn(O, E).
hasParticipant(E, O) :- isParticipantIn(O, E).

Communities instantiated:

member(M) :- magistrate(M).
community(C) :- group(C).
isMemberOf(M, C) :- community(C), member(M), belongs(M, C).
hasMember(C, M) :- isMemberOf(M, C).
```

Figure 3.2: Co-Participation, Participation and Communities ODPs instantiated.

The modeling of other patterns in FOL can be found in Appendix A.1.

## 3.2 The PRETHY (Pre-REvision THeorY)

In this section, the approach for pattern-based ontology evolution through theory revision and ODPs is described. When ontologies are developed axiomatically, systematically and formally, they are called logical theories [2]. So, is discussed how the application of theory revision techniques to automatically evolve an ontology, built according to ODPs can benefit from the use of these patterns. Since ODPs are a set of good practices that can be used for construction, maintenance, interoperability, understanding by the user, among other, for the purpose of getting a product of good quality, they can be treated as background knowledge. In this condition, they should be protected from the revision process, because they are considered correct. Figure 3.3 describes the proposal of this dissertation.

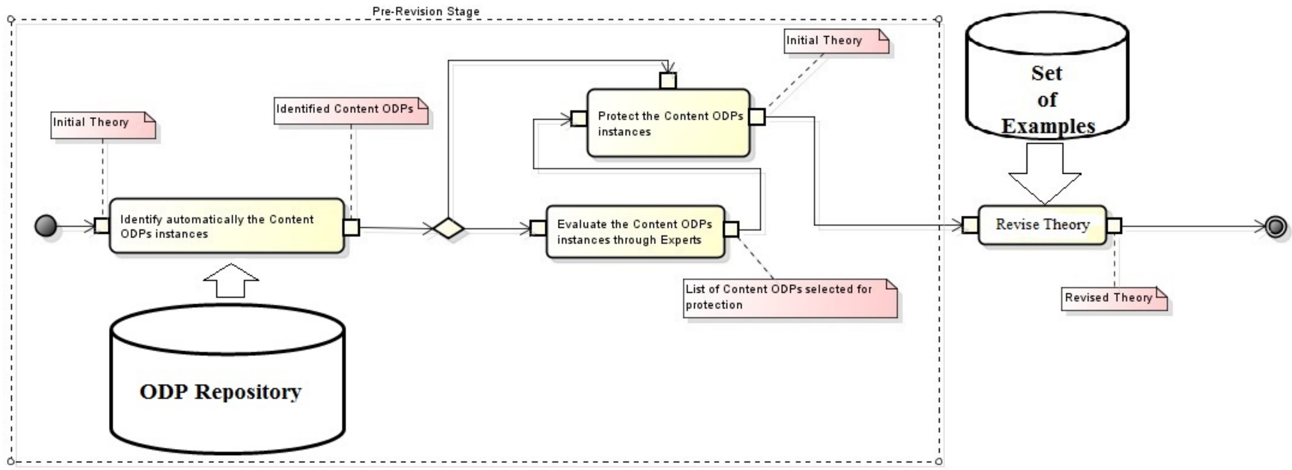


Figure 3.3: Proposal's Schema

The pre-revision process begins with the automatic identification of ODPs step. It receives as input the theory ($THY \cup FDT$) and the repository of ODPs (see Figure 3.1), as shown in Figure 3.3. In THY, the ODPs are found instantiated, so the pre-revision process tries to find the registered ODPs in the ODP Repository in the THY, through an exhaustive search. This search is made looking for rules of the ODPs in THY. As seen in section 3.1, these rules contain predicates of ODPs and predicates of domain. Initial identification is made through the predicates of ODPs and then try to identify exacts or approximate occurrences of ODPs through of the domain's predicates according to the structure of the rule defined to the ODPs in the repository. An occurrence is said to be exact when the ODP found in THY has identical structure to the one defined in the repository. If the structure is similar (absence of any defined element or presence of an unknown element), the occurrence is said to be approximate. The ODPs instances found form a subset of THY's clauses here called $THY_P$.

After obtaining the $THY_P$ and according to the selected option, the pre-revision process can take one of the following decisions:

1. Consider all $THY_P$ as background knowledge and protects it, by including $THY_P$ in FDT.
2. Submit $THY_P$ to a expert who will decide which clauses should be part of the background knowledge.

To illustrate the first option, consider that the THY contains, among others, the following clauses concerning the travel domain:

1. member(C) :- contact(C).
2. community(G) :- group(G).
3. isMemberOf(C, G) :- community(G), member(C), belongs(C, G).
4. hasMember(G, C) :- isMemberOf(C, G).
5. object(O) :- contact(O).
6. event(E) :- activity(E).
7. isParticipantIn(O, E) :- event(E), object(O), gender(O, male), participates(O, E).
8. hasParticipant(E, O) :- isParticipantIn(O, E).
9. hasAccommodation(D, A) :- isAccommodationOf(A, D).
10. hasActivity(D, A) :- isActivityOf(A, D).
11. isOfferedAt(A, D) :- hasActivity(D, A).
12. hasPart(D1, D2) :- isPartOf(D2, D1).
13. hasPart(D1, D2) :- isPartOf(D1, X), hasPart(X, D2).
14. hasContact(A, C) :- isParticipantIn(C, A).
15. hasRating(A, R) :- isRatingOf(R, A).
16. isHotelAccommodation(H, A) :- hotelAccommodation(H, A).
17. isCampgroundAccommodation(C, A) :- campgroundAccommodation(C, A).
18. isBedAndBreakfastAccommodation(B, A) :- bedAndBreakfastAccommodation(B, A).

The $THY_P$ set generated during the pre-revision process contains the clauses (1) to (8). After generating this set, the process continues and all $THY_P$ is forwarded to protection step.

In the second option, the set of clauses ($THY_P$) also contains the clauses(1) to (8). It is then presented to an expert, which according to expertise and positive examples like hasContact(sports, joana), decides that only communities ODP, clauses (1) to (4), will be protected. This is because, in this case, the participation ODP, clauses (5) to (8), was defined as being mandatory that the contact is male. And according to the positive example described above, the participation has no obligation to be a man. So, only the set of clauses selected by him is sent to protection step.

The last step of the pre-revision process, in both options, is the protection of the selected clauses. This step receives a set of clauses ($THY_P$) to be moved from THY to the FDT to generate FDT'. The THY is also updated to generate THY'. So, FDT' and THY' are forwarded to the revision process that will generate THY", as described in section 2.3.

The theory revision process transforms an initial incorrect or incomplete theory in a revised theory that correctly represents the domain of knowledge. In proposal of this dissertation, it is conducted through the FORTE system [24] as described in Algorithm 1. The revised theory can represent the transformation of the initial theory. This transformation always occurs when the initial theory is improved by the process.

## 3.3 Analysis of Revision Space

According to Richards and Mooney [24], a theory revision process contains two search spaces: revision space and solution space. The revision space is formed by the modifiable area of theory (THY). For the foundation of this dissertation, the revision space (THY) is composed by clauses of design patterns ($THY_P$) and other clauses ($THY_{NP}$) that may or may not reference any design pattern. Formalizing this foundation:

$$THY = (THY_P \cup THY_{NP}), \text{ where } THY_P \neq \phi \text{ and } THY_{NP} \neq \phi \text{ and } (THY_P \cap THY_{NP}) = \phi$$

The condition $THY_P \neq \phi$ is a necessary prerequisites for the proposal presented in this work, because if it is empty the result of the revision will be equal to the input. The revision space also contains the set of revision points (R), which may intersect with the $THY_P$ and $THY_{NP}$ subsets. This revision space (THY) can be originally defined, before the revision process, as described in the Tables 3.1, 3.2 and 3.3 that show the possible configurations of THY given R.

| |
|---|
| member(C) :- contact(C). |
| community(G) :- group(G). |
| isMemberOf(C, G) :- community(G), member(C), belongs(C, G). |
| hasMember(G, C) :- isMemberOf(C, G). |
| object(O) :- contact(O). |
| event(E) :- activity(E). |
| isParticipantIn(O, E) :- event(E), object(O), participates(O, E). |
| hasParticipant(E, O) :- isParticipantIn(O, E). |
| hasAccommodation(D, A) :- isAccommodationOf(A, D). |
| hasActivity(D, A) :- isActivityOf(A, D). |
| isOfferedAt(A, D) :- hasActivity(D, A). |
| hasPart(D1, D2) :- isPartOf(D2, D1). |
| hasPart(D1, D2) :- isPartOf(D1, X), hasPart(X, D2). |
| hasContact(A, C) :- isParticipantIn(C, A), **gender(C, male)**. |
| hasRating(A, R) :- isRatingOf(R, A). |
| isHotelAccommodation(H, A) :- hotelAccommodation(H, A). |
| isCampgroundAccommodation(C, A) :- campgroundAccommodation(C, A). |
| isBedAndBreakfastAccommodation(B, A) :- bedAndBreakfastAccommodation(B, A). |

Table 3.1: Example of revision point in the non-pattern clause.

In the Table 3.1, the light gray frame comprises clauses in $THY_P$ and the gray frame comprises clauses in R. The clause *hasContact(A,C)* is wrong when the input set of facts include female participants. In this case, the framed clause (*hasContact*) belonging to the set of non-pattern clauses ($THY_{NP}$) also contains a revision point that belongs to the set of revision points (R), that is highlighted in bold.

| |
|---|
| member(C) :- contact(C). |
| community(G) :- group(G). |
| isMemberOf(C, G) :- community(G), member(C), belongs(C, G). |
| hasMember(G, C) :- isMemberOf(C, G). |
| object(O) :- contact(O). |
| event(E) :- activity(E). |
| isParticipantIn(O, E) :- event(E), object(O), **gender(O, male)**, participates(O, E). |
| hasParticipant(E, O) :- **isParticipantIn(O, E)**. |
| hasContact(A, C) :- **isParticipantIn(C, A)**. |
| hasAccommodation(D, A) :- isAccommodationOf(A, D). |
| hasActivity(D, A) :- isActivityOf(A, D). |
| isOfferedAt(A, D) :- hasActivity(D, A). |
| hasPart(D1, D2) :- isPartOf(D2, D1). |
| hasPart(D1, D2) :- isPartOf(D1, X), hasPart(X, D2). |
| hasRating(A, R) :- isRatingOf(R, A). |
| isHotelAccommodation(H, A) :- hotelAccommodation(H, A). |
| isCampgroundAccommodation(C, A) :- campgroundAccommodation(C, A). |
| isBedAndBreakfastAccommodation(B, A) :- bedAndBreakfastAccommodation(B, A). |

Table 3.2: Example of revision points in the THY.

In the Table 3.2, the light gray frame comprises clauses in $THY_P$. The clause *isParticipantIn(O, E)* is wrong when the input set of facts include female contacts. The set of revision points (R) is formed by revision points, highlighted in bold. So the set of revision points comprises pattern and non-pattern clauses. The gray frame comprises the non-pattern clause in R (*hasContact(A, C)*).

| |
|---|
| member(C) :- contact(C), **gender(C, male)**. |
| community(G) :- group(G). |
| isMemberOf(C, G) :- community(G), **member(C)**, belongs(C, G). |
| hasMember(G, C) :- **isMemberOf(C, G)**. |
| object(O) :- contact(O). |
| event(E) :- activity(E). |
| isParticipantIn(O, E) :- event(E), object(O), participates(O, E). |
| hasParticipant(E, O) :- isParticipantIn(O, E). |
| hasAccommodation(D, A) :- isAccommodationOf(A, D). |
| hasActivity(D, A) :- isActivityOf(A, D). |
| isOfferedAt(A, D) :- hasActivity(D, A). |
| hasPart(D1, D2) :- isPartOf(D2, D1). |
| hasPart(D1, D2) :- isPartOf(D1, X), hasPart(X, D2). |
| hasContact(A, C) :- isParticipantIn(C, A). |
| hasRating(A, R) :- isRatingOf(R, A). |
| isHotelAccommodation(H, A) :- hotelAccommodation(H, A). |
| isCampgroundAccommodation(C, A) :- campgroundAccommodation(C, A). |
| isBedAndBreakfastAccommodation(B, A) :- bedAndBreakfastAccommodation(B, A). |

Table 3.3: Example of revision points in the pattern clauses.

In the Table 3.3, the light gray frame comprises clauses in $THY_P$. The clause *member(C)* (which is part of the occurence of the Community ODP) is wrong when the input set of facts include female members. Thus, the set of revision points (R) is composed only revision points in the set of pattern clauses, highlighted in bold.

According to the proposal of this dissertation, when the protection stage is automatic, the specialist is not consulted to define which ODP clauses should be protected and, in this case, all the pattern clauses should be moved to the FDT. Therefore, the revision space is changed (THY') and shall contain only non-pattern clauses (THY' = $THY_{NP}$). This modification of the revision space may impact the generation of the set of revision points, as will be shown in the following section.

From the definition of the revision space (THY'), was verified that the set of fault points of the theory can be modified and thus impact the generation of the set of revision points called R'. A fault point is a point on the theory that can generate several revision points, depending on the set of examples. The set of fault points has a relation with the set R defined in the previous section, as shown below:

1. $R' = R$

2. $R' \subset R$

3. $R' = \phi$

The scenario (1) occurs when part of the set of fault points is moved to the FDT and the other part remains in the revision space (THY). For scenario (2), the set of fault points has been completely moved to the FDT. When the scenario (3) occurs, the set of fault points remains entirely within the revision space (THY).

The scenarios (1) and (2) mentioned above are simulated in experiments, and it was observed that the proposal of this dissertation can result in changes in the set of revision points. The proposal's impacts identified in experiments are analyzed.

# 4. Experimentation

This experiment aims to evaluate the impacts on the revision space defined in section 3.3, of this proposal, to confirm the hypothesis of this dissertation. In this way, theories of the family law domain will be used, because this domain has characteristics of the ontologies considered in this dissertation: large, complex and can use ODPs to obtain high quality. They were constructed specially from observations made in the current Brazilian Civil Code, available in PDF format on the Library of Brazilian National Congress Senate site[1] , for this work. On 05.05.2011, the Federal Supreme Court (STF) decided that civil marriages of same-sex couples are valid. The National Council of Justice (CNJ)[2] reinforced this decision with the Resolution nº 175 of 05.14.2013. This decision would result in the need for revision of a logical theory that would represent the Brazilian Civil Code, because this defined the civil marriage as an union between a man and a woman.

## 4.1 Experimental Methodology

To confirm the hypothesis, a controlled experiment was performed where theories, examples and background knowledge defined in Appendices A.2, A.4 and A.5, respectively were used and failure points added manually in the theories. The theory of Appendix A.2.1 was manually constructed from Brazilian Civil Code and prepared to evaluate the impacts of a scenario where $R = R'$. Nonstandard clauses (41 and 42) should be generalized, because some of its antecedents prevent the realization of civil marriages between same-sex couples. The theory of Appendix A.2.2 is incomplete and was manually constructed from decision of Brazilian Supreme Court and prepared to evaluate if the proposal of this dissertation prevents that correct and complete ontologies are obtained. This theory allows civil marriages between same-sex couples and must be specialized to only allow

[1] http://www2.senado.leg.br/bdsf/bitstream/handle/id/70327/C%C3%B3digo%20Civil%202%20ed.pdf?sequence=1

[2] http://www2.stf.jus.br/portalStfInternacional/cms/destaquesNewsletter.php?sigla=newsletterPortalInternacionalDestaques &idConteudo=238515

civil marriages between couples of different sex. In this case, were not inserted points of failure in theory, but it was considered that a change of context has become incomplete. The theory of Appendix A.2.3 was manually constructed from Brazilian Civil Code and prepared to evaluate the impacts of a scenario where $R' \subset R$. Patterned clause (7) should be generalized, if possible, because it has an antecedent that does not belong to the ODP. The scenario $R' = \phi$ will not be evaluated because there is no revision points in the revision space and therefore the initial ontology will always be equal to ontology revised. The Table 4.1 shows initial characteristics of revision spaces used in this experiment.

| | THY A.2.1 | THY A.2.2 | THY A.2.3 |
|---|---|---|---|
| Number of clauses | 42 | 41 | 40 |
| Number of predicates (total) | 135 | 120 | 121 |
| Average number of antecedents per clause | 2.1 | 2.0 | 2.0 |

Table 4.1: Characteristics of the theories

The examples (216) were created based on these theories to train and test them. These examples evaluate the following relationships:

- civilMarriage(H, W, J, E).
- coParticipatesWith(H, W, E).

CivilMarriage was defined as an n-ary relationship between the couple (H and W), a magistrate (J) and an event (E), which is the matrimony itself. The newlyweds are instances of the concept of person, the magistrate is an instance of this concept and the event is an instance of the concept of matrimony. CoParticipatesWith is a clause of co-participation ODP that works with the concepts of person and event. The newlyweds are instances of the concept of person and the event is an instance of the concept of matrimony.

The background knowledge consists of facts (2723) used the examples and others who help organize the experiment, defining some kinship relations and forming families. The facts are instantiations of concepts and relationships defined in the ontology and beyond those mentioned during the explanation of the examples, the following concepts and relationships are part of the background knowledge: gender, magistrate, magistracy, belongs, group, single, adult, adolescent, killed, adopted, civilkinship, progenitor, havePermissionToMarry, married, matrimony and participatesIn.

In the experiment, each theory is submitted to three revision processes: only performed by FORTE system, simulating the proposal of this dissertation in automatic mode and simulating the proposal of this dissertation in semiautomatic mode. For each revision process, the set of examples is formed and background knowledge selected according to the examples chosen.

The data collected for analysis is the following: number of revision points, number of proposed revisions, initial size of THY, final size of THY, number of predicates moved to the FDT, initial accuracy of the theory and final accuracy of theory. The analysis on these data was done by performing comparisons between the data collected from a theory revision system using its current form (that is, without distinguishing ODP occurrences in the THY) and the data collected using the proposal of this dissertation, that is, considering the ODPs as background knowledge. All data collected in controlled experiment is organized in 3 column tables. The 1st column contains the data collected from the FORTE system, the 2nd column contains the data collected based on the PRETHY option that considers all ODPs instantiated in a theory as background knowledge (automatic mode) and the 3rd column contains the data collected based on the PRETHY option where an expert selects the ODPs that should be considered as background knowledge (semiautomatic mode).

Ratings in all scenarios were performed using an HP Pavilion dm4-2155br notebook with the following configuration: Windows 7 operation system, Intel Processor I5 (2.4 GHz), RAM Memory of 4 GB and one Hard Drive of 640 GB.

## 4.2 Threats to the Validity of the Experiment

A set of examples poorly constructed may represent a threat to internal validity of the experiment, because it may induce the experiments to unsatisfactory results. This threat is related to the paradigm of inductive learning that uses training data (examples) to perform its task and has as disadvantage of generating unsatisfactory results if the training data does not fully represent the domain, especially difficult for large and complex ontologies. This learning paradigm is used in machine learning that is a field of multidisciplinary research, while focused on solving the problem of building computer programs able to improve their performance in the execution of their tasks, using examples that related to them (tasks) [30].

This threat is associated with the proposal of this dissertation, because Theory Refinement (see 2.3 for more details) is a subarea of Inductive Logic Programming (ILP). This in turn is the intersection of machine learning and logic programming, that acts in construction and maintenance of theories, by applying techniques of inductive learning and logic programming [31, 32]. So, Theory Revision, which is a task of Theory Refinement, is subject to this kind of threat.

Threats to external validity of an experiment are the threats on the confidence that the results are generalizable. This experiment uses ontologies domain of family law, so steps should be taken so that the results obtained here are valid for other domains, in other words, the results will be repeated for other domains.

## 4.3 Guarantee of the Validity of the Experiment

To ensure the internal and external validity of the experiment, some actions were taken. For the examples, internal validity, the actions taken are: reducing the problem of domain representation and expressiveness of examples created. The adoption of the revision theory to evolve ontologies reduces the need for the set of examples of completely represent the domain, because it is focused almost correct or complete ontologies and thus the set of examples should represent only the knowledge that need to evolve of ontology and not all the domain. The expressiveness of the examples is making guaranteed that the set of examples containing all possible combinations for civil marriage: between men and women, among women and among men, and ensure other necessary conditions for holding a marriage. The evaluation is then performed in 10 steps, where in each step, 9 sets are used for training and 1 set for testing, so that all sets have been used once as test set (tenfold cross-validation [33]).

Regarding external validity, which guarantees that the results of the experiment are generalizable, the following measures were adopted: conducting independent experiments and performed on the two options of the proposal of this dissertation (automatic mode and semiautomatic mode). The three experiments in this section were performed independently, meaning there are no dependencies and interferences between them, so that the results of an experiment does not interfere with others. To ensure this independence each experiment has its own logical theory, which is not generated by other (revised theory), set of examples (different from others), as well as the background knowledge. As the experiments are controlled, where the errors are inserted purposefully, they can be seen as artificial, different from a real situation. To avoid this, all experiments evaluate automatic and semiautomatic modes of the proposal of this dissertation. In this case, the evaluations of the options in each experiment uses the same logical theory, the same set of examples and the same background knowledge.

## 4.4 Results of Experiment

In first evaluation was considered the logical theory described in Appendix A.2.1 and in this theory only marriages between persons of different sex are allowed. In one scenario, according to the decision of the Supreme Court, where marriage between persons of the same sex is allowed, this theory must be generalized. The set of examples is the same in all revision processes realized and is formed by positive examples representing civil marriage between different sex couples and same-sex couples, the negative examples represent impediments to the realization of civil marriage, regardless of the sex of the couple. Table 4.2 shows the results obtained by the revision processes to which the theory was submitted.

| | FORTE system | PRETHY System (Automatic mode) | PRETHY System (Semi-automatic mode) |
|---|---|---|---|
| Number of generalization points found | 22 | 22 | 22 |
| Number of specialization points found | 0 | 0 | 0 |
| Number of proposed revisions | 13 | 11 | 12 |
| Initial THY size | 135 | 112 | 125 |
| Final THY size | 133 | 110 | 123 |
| Number of predicates moved to the FDT | 0 | 23 | 10 |
| Initial Test Accuracy | 66.67% | 66.67% | 66.67% |
| Final Test Accuracy | 100.00% | 100.00% | 100.00% |

Table 4.2: Results table of revisions for logical theory in Appendix A.2.1

The processes described in Table 4.2 performed the same revision, which is shown below:

From:

1. civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D), canMarry(A,B,D).
2. civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D), canMarry(A,B,D).

To:

1. civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
2. civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D), canMarry(A,B,D).

All revision process executed generated 22 revision points and reached a theory's final accuracy of 100%. However, despite implementing the same revision to correct the initial theory, some differences were identified between the process. The first is the number of revisions proposed by each process. PRETHY in automatic mode proposed unless revisions (11),followed by PRETHY in semiautomatic mode (12) and finally the FORTE system (13). This indicates that the proposal of this dissertation was more efficient than a traditional revision process represented by FORTE system. The complete lists of proposed reisions for each revision process can be seen in Appendices A.3.1, A.3.2 and A.3.3 for the FORTE system, PRETHY in automatic mode and PRETHY in semiautomatic mode, respectively. The second was the initial and final revision space (THY) sizes, which represent the number of predicates in this space. In FORTE system, all clauses of ODPs were maintained in THY, PRETHY in automatic mode all the clauses (1) to (9) were moved to the background knowledge (FDT) and PRETHY in semiautomatic mode, the clauses (1) to (4) were moved to the FDT. Consequently, the number of predicates moved to the FDT, which represents the total number of predicates of the clauses moved to it is bigger in the PRETHY than in the FORTE system. Thus, the numbers indicate that the proposal of this dissertation was able to correct the same initial theory through of an minor revision space.

For the second evaluation performed to ensure that the proposal of this dissertation does not prevent that a correct and complete ontology is obtained, was considered the logical theory described in Appendix A.2.2. This theory allows marriage between persons of different sex and between persons of the same sex. In a scenario where only couples of different sex can marry, this theory is incorrect and needs to be specialized. This context change could happen, for example, through a constitutional amendment suggested by conservative parties that veto the legal union of homosexual couples. Table 4.3 shows the results of the revision process conducted in this scenario.

| | FORTE system | PRETHY System (Automatic mode) | PRETHY System (Semi-automatic mode) |
|---|---|---|---|
| Number of generalization points found | 0 | 0 | 0 |
| Number of specialization points found | 13 | 6 | 10 |
| Number of proposed revisions | 24 | 12 | 18 |
| Initial THY size | 120 | 97 | 110 |
| Final THY size | 127 | 104 | 117 |
| Number of predicates moved to the FDT | 0 | 23 | 10 |
| Initial Test Accuracy | 57.14% | 57.14% | 57.14% |
| Final Test Accuracy | 100.00% | 100.00% | 100.00% |

Table 4.3: Results table of revisions for logical theory in Appendix A.2.2

The processes described in Table 4.3 performed the same revision, which is shown below:

From:

- civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D).

To:

- civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(B,male), gender(A,female).
- civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(A,male), gender(B,female).

Through observation of Table 4.3, was verified that the revision processes executed behaved in different manner, despite implementing the same revision to correct the initial theory, only the theory's final accuracy obtained by them was the same: 100%. The number of revision points generated are different: for FORTE system 13, for PRETHY in automatic mode 6 and for PRETHY in semiautomatic mode 10. This indicates a better performance of the PRETHY in generating revision points. The numbers of proposed revisions presented reduction between FORTE system (24) and the PRETHY in semiautomatic mode(18) and this with PRETHY in automatic mode (12), which indicates that the proposal of this dissertation was also more efficient in this situation. The complete lists of proposed revisions for each revision process can be seen in Appendices A.3.4, A.3.5

and A.3.6 for the FORTE system, PRETHY in automatic mode and PRETHY in semiautomatic mode, respectively. The initial and final revision space (THY) sizes are different too. In FORTE system, all clauses of ODPs were maintained in THY, PRETHY in automatic mode all the clauses (1) to (9) were moved to the background knowledge (FDT) and PRETHY in semiautomatic mode, the clauses (1) to (4) were moved to the FDT. Consequently, the number of predicates moved to the FDT is bigger in the PRETHY than in the FORTE system. Thus, the numbers indicate that the proposal of this dissertation was able to correct the same initial theory through of an minor revision space.

For the last evaluation was considered the logical theory described in Appendix A.2.3. Here, it is intended to evaluate the impacts on the scenario in which R' is a subset of R ($R' \subset R$), which occurs in the revision process performed by PRETHY in automatic mode. Thus, considering that the newlyweds do not need to invitation for your own marriage and the models presented in section 3.1 for participation and co-participation patterns, the *isInvited* antecedent of clause (7) of the theory may be a fault point. Therefore, this theory must be generalized and Table 4.4 shows the results obtained in the revision process used.

| | FORTE system | PRETHY System (Automatic mode) | PRETHY System (Semi-automatic mode) |
|---|---|---|---|
| Number of generalization points found | 6 | 6 | 6 |
| Number of specialization points found | 0 | 0 | 0 |
| Number of proposed revisions | 9 | 2 | 9 |
| Initial THY size | 121 | 97 | 111 |
| Final THY size | 120 | 101 | 110 |
| Number of predicates moved to the FDT | 0 | 24 | 10 |
| Initial Test Accuracy | 40.00% | 50.00% | 40.00% |
| Final Test Accuracy | 100.00% | 1000.00% | 100.00% |

Table 4.4: Results table of revisions of logical theory in Appendix A.2.3

In this case, the process in Table 4.4 did not perform the same revision. The PRETHY in automatic mode yielded the following revision:

From:

- civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D).

To:

- civilMarriage(A,B,C,D):- participatesIn(C,D),participatesIn(B,D),participatesIn(A,D).
- civilMarriage(A,B,C,D):- allowedMarriage(C,D),canMarry(A,B,D).

The revisions proposed by the FORTE system and PRETHY in semiautomatic mode processes of Table 4.4 were identical and are shown below:

From:

- isParticipantIn(A,B):-object(A),event(B),fdt:participatesIn(A,B),isInvited(O, E).

To:

- isParticipantIn(A,B):-object(A),event(B),fdt:participatesIn(A,B).

From revisions obtained, was verified that the best option was to keep the ODP considered incorrect in the revision space, thus allowing it to be revised. Both FORTE system and PRETHY in semiautomatic mode were able to fix the incorrect clause. PRETHY in automatic mode removed all the ODPs of the revision space can not be considered an appropriate process, because in cases like this, where the failure points were protected, it can not fix them properly, despite getting a revised theory with 100% accuracy. See the following clauses:

- allowedMarriage(J, E) :- isParticipantIn(J, E), isMemberOf(J, tj).
- canMarry(H, W, E) :- coParticipatesWith(H, W, E), ableToCivilMarriage(H, W).
- civilMarriage(H, W, J, E) :- canMarry(H, W, E), allowedMarriage(J, E).

The above clauses mean that a civil marriage may take place if the bride and groom participate in the marriage, they are able to marry and a judge perform the ceremony. The revision proposed by PRETHY in automatic mode requires only the bride, groom and a third element participating in the marriage, not guaranteeing the other conditions necessary for conducting the ceremony. So, it is not an appropriate solution.

The number of revision points generated in three revision processes was the same:6. The FORTE system and the PRETHY in semiautomatic mode generated the same set of revision points and although the set of revision points generated by the PRETHY in automatic mode be a subset of first set, some revision points of it are not found in the first set. This may mean that the PRETHY in automatic mode was able to go through the revision space of different way of the FORTE system and of the PRETHY in semiautomatic mode, finding revision points disregarded by them. The difference may reside in the use of different sets of examples to revise the thory. As for the PRETHY in automatic mode ODPs are all protected, in this case also the failure point is protected and thus it was necessary to use a different set of examples of used by the FORTE system and PRETHY in semiautomatic mode, which can maintain the failure point within revision space. The PRETHY in automatic mode used the set of examples formed by examples of the type *civilMarriage(H, W, J, E)*, while FORTE system and PRETHY in semiautomatic mode used the set of examples of the type *coParticipatesWith(H, W, E)*.

All revision processes were able to obtain a theory's final accuracy of 100%, although the PRETHY in automatic mode has not had a satisfactory performance in this aspect as shown above. The number of proposed revisions by the FORTE system and PRETHY in semiautomatic mode was identical: 9. While PRETHY in automatic mode generated only 2 proposed revisions. Although it seems more efficient than the FORTE system and PRETHY in semiautomatic mode, the PRETHY in automatic mode is not more efficient as verified through accepted revision by him and PRETHY in semiautomatic mode has the same efficiency of the FORTE system. The complete lists of proposed revisions for each revision process can be seen in Appendices A.3.7, A.3.8 and A.3.9 for the FORTE system, PRETHY in automatic mode and PRETHY in semiautomatic mode, respectively. The initial and final revision space (THY) sizes are different too. In FORTE system, all clauses of ODPs were maintained in THY, in PRETHY in automatic mode all the clauses (1) to (9) were moved to the background knowledge (FDT) and in PRETHY in semiautomatic mode, the clauses (1) to (4) were moved to the FDT. Consequently, the number of predicates moved to the FDT is bigger in the PRETHY than in the FORTE system. It s important to observe that the final size of THY revised by PRETHY in automatic mode is larger than the PRETHY in semiautomatic mode. In this context, the proposal of this dissertation was able to adequately correct the initial theory, reducing the revision space only through the semiautomatic mode.

### 4.5 Remarks about Experiment

The objective of this controlled experiment was to evaluate the impacts of the proposal of this dissertation over ontology evolution process. It was important to understand how it changed the process to identify strengths and weaknesses and evaluate the viability of proposal of this dissertation. For this, some scenarios were evaluated, varying the dependent variables considered important in evaluation of the proposal of this dissertation.

These scenarios are those where the set of revision points is not moved, is partially or is all moved to the FDT. In other words, either the set of revision points is entirely in the THY, it was partially or totally moved to the FDT.

Thus, some scenarios was evaluated: a scenario where the set of revision points is in the THY ($R = R'$) represented by a generalization process, a scenario where the set of revision points is defined during the process represented by a specialization process that is not linked directly to any scenario of section 3.3 and was performed to prove that the proposal of this dissertation is really able to get correct and complete ontologies and a scenario where the set of revision points was partially moved to the FDT ($R' \subset R$) represented by another generalization process. Each scenario was evaluated using the proposal of this dissertation in its two modes and the FORTE system. The collected data was the same to avoid different evaluations that would invalidate the experiment and to allow comparison among scenarios. So that, it was possible to compare the proposal of this dissertation and an traditional revision process (FORTE system) and to define when should be used each option of the proposal of this dissertation.

# 5. Related Works

In this chapter, some related works with the proposal of this dissertation are presented and comparisons between them are made.

In Coulet et al [16], a proposal for a semiautomatic refinement of ontologies in Description Logic (DL) [34] is described. This proposal is based on a process RAA (Role Assertion Analysis) [16] and consists of four steps:

- generate a set of graphs from instances of a concept.
- transform the graphs in a formal context.
- analyse the formal context using FCA (Formal Context Analysis).
- transform the FCA results on new DL concepts and relations.

The graph mentioned above is directed and cyclic, its root is the instance of the concept, the vertices are instances of attributes and edges instances of relations. These graphs are explored by a depth-first search algorithm with deep limiter to find new relationships between individuals of the ontology. Formal context considers the association of the set of instances of a concept with a set of attributes through a relationship. The FCA abstracts conceptual descriptions from a set of objects described by attributes.

The proposal in [16] does not consider the ODPs to optimize the process, and may invalidate the use of these, according to the decision taken. The proposal of this dissertation seeks to preserve the ODPs of the suggested modifications and to optimize the process.

A proposal for refinement of ontologies, independent of the representation language, performed by complementing partial instances of ontology design patterns semiautomatic way, is proposed in [17]. The central idea is to look for components within the ontology that partially instantiate some ODP. Instantiations of ODPs are identified by their structure and meaning of its elements, expressed by the axioms and lexical features of each element.

Once identified a partial instantiation of an ODP, it can identify missing elements and generate a list of suggestions for refinement.

The main difference between the proposal in [17] and the proposal of this dissertation lies in the fact that the first seek to correct an ontology suggesting the adoption of new ODPs in the existing structure of a semiautomatic way, which can be costly, time consuming and error prone. While the proposal of this dissertation, the main objective is not to allow the existing ODPs are changed, which can be done automatically reducing or eliminating the disadvantages of the proposal in [17].

A method for automatic revision of ontologies in OWL (Ontology Web Language) is the proposal of [10]. In the proposal, the ontology in OWL is converted into a logical theory in first-order logic (FOL) to then be revised by the FORTE system. The outcome of the revision, the revised theory, is then converted back into an refined ontology in OWL. However, this work does not take into account the design patterns that can be part of the initial ontology. Indeed, in the conversion process itself, existing ODPs can be lost.

The proposal in [10] and the proposal of this dissertation can act in a complementary fashion, with the first acting mainly in the conversion of ontologies in OWL for ontologies in FOL and vice versa (after revision) and the second performing the revision process itself. But to keep the purpose of the proposal of this dissertation, the conversions performed must meet the ODPs.

Rudolph et al [11] propose a way of ontology creation and refinement combining techniques in natural processing language (NLP) and formal concept analysis (FCA), focused on large and complex ontologies. The NLP techniques are used to identify and suggest knowledge elements from textual sources important for the domain. And FCA techniques are used to provide guidance during the knowledge specification process.

Besides not considering the ODPs during the refinement process, the proposal in [11] starts from the premise that there is no background knowledge and also is not able to perform minimal modifications in the ontology, which leads to relearn all the ontology. These characteristics differentiates the proposal of this dissertation that besides considering the ODPs is able to perform minimal modifications to the original ontology.

Text-2-Onto [12] is the framework used for learning and refinement of ontologies from texts, but it does not guarantee that the generated model is accurate and correct. To overcome this deficiency, relational exploration techniques are used. These techniques are based on attribute exploration algorithm from FCA. The same considerations made for the proposal in [11] regarding the proposal of this dissertation can be applied to this.

Corcho et al [13] propose guidelines for ontology debugging based on anti-patterns. The proposal tries to solve the problem of ontology debugging tools that are not able to explain efficiently and correctly the main causes of a concept to be unsatisfactory, combined with little support tools to propose solutions. These ontologies must be expressed in OWL-DL, possibly developed by domain experts who do not know deeply DL and hence can misuse the constructors of DL and misunderstand the semantics of some OWL expressions. In this context, the domain experts were changing the original ontology axioms randomly, altering the intended meaning of the definitions instead of correcting the errors of their formalization.

The proposal is based on the identification of anti-patterns and a debugging strategy which can be combined with existing tools. Anti-patterns are defined as ineffective patterns or far from ideal, which is the worst practice to design and build software [13]. A set of anti-patterns generally used by domain experts and that result in modeling errors or unsatisfactory concepts were identified, they are: Detectable Logical Anti-Patterns (DLAP), Cognitive Logical Anti-Patterns (CLAP) and Guidelines (G). DLAP represent what DL reasoners and debugging tools normally detect. CLAP represent possible modeling errors that may be due to a misunderstanding of the logical consequences of the used expression. G represent complex expressions used in an ontology component definition that are correct from the logical and cognitive points of view, but for which the ontology developer could have used other simpler alternatives or more accurate. The debugging strategy is to identify and solve anti-patterns of the easiest to the most difficult, to these with the aid of a debugging tool and removing superfluous axioms.

The proposal in [13] is aimed at correcting problems from the possible lack of experience of a domain expert with the formalism (DL) used in the ontology. However, it is not focused on the evolution of concepts and relationships that can occur over time, as the proposal of this dissertation.

Friedrich and Shchekotykhin [14] make a proposal for debugging ontologies expressed in OWL-DL. This proposal consists of two stages: diagnosis and repair. The diagnosis stage is the elaboration of the test cases, by the knowledge engineer, to define the minimal set of axioms which, if corrected, make consistent ontology. A consistent ontology is one that meets the requirements defined by the knowledge engineer. The repair stage consists in fix itself of the ontology through two basic operations: deletion and addition of axioms. So, reparation of axioms includes the exclusion of incorrect axioms and addition of corrected axioms, so that at the end of the ontology is consistent. Exclusions and additions of axioms are performed without ensuring that ODPs are not changed, though a well specified ODP does not contain inconsistent axioms.

Debugging an ontology can be a difficult task in which the requirements and test cases must be specified for the target ontology be obtained. To meet them, all axioms belonging to a diagnosis should be changed. However, existing diagnosis methods may return multiple diagnoses for the set of requirements and test cases. In this case, the task of selecting the most appropriate diagnosis lies with the user. In [15], one approach is proposed to reduce the set of diagnoses getting additional information through queries made to users, a information extraction system, etc. Each answer provided, the set of diagnoses is reduced until the diagnosis target be obtained. To construct the queries, the proposal is based on the fact that different diagnoses are formed by different sets of axioms. Thus, the queries define whether an axiom should or not participates in the target ontology. This proposal tries to obtain the target diagnosis among several possible according to the requirements and test cases defined. It does not effectively corrects an ontology limiting itself to prepare a diagnosis that identifies the set of axioms that must be changed to achieve the target ontology. It also does not guarantee that the ODPs instantiated in an ontology not participated in the target diagnosis for subsequent changes. When generating the diagnostics and selection of the correct diagnosis, ODPs are not considered. The ODPs are composed of competency questions, which could be part of the test cases used to get the correct diagnosis for the target ontology, there is no obligation.

A system that generates explanations in English because of an entailment is due to an ontology is proposed in [18]. These explanations are used to assist developers with little knowledge in OWL in the repairing of wrong entailments. To generate these explanations, the system starts a justification that is linked to the entailment by a proof tree constructed from a set of deduction rules. The root node of this tree is the entailment, the terminal nodes are the axioms in the justification and the other nodes are intermediate statements introduced by a deduction rule. This is an inferential step from the axioms of justification for the entailment. A Justification is defined as a subset of the ontology from which the entailment can be drawn. They may are several justifications for an entailment and also several proof trees for each justification. For each proof tree is assigned values to all its rules to then be possible to estimate the difficulty of it and then select one. The selected proof tree is then used to generate an explanation in English. Although the ODP well specified do not generate wrong entailments, there is no guarantee that they will not be modified as the proposal of this dissertation can do.

In [19] is proposed an ontology revision process that makes the quality control for ontological knowledge acquired automatically. The proposal assume that the set of axioms deducted from confirmed axioms (deductive closure) must be disjoint of the set of declined axioms to be able to automate partially the ontology revision process through the use of a reasoning tool. From decisions taken by expert the axioms not evaluated can be deleted or inserted, depending on their logical relationships with the axioms already evaluated. To reduce the need for reasoning operations, a decision space contains a track of the dependencies between the axioms. This approach is not associated with any formalism, requiring only that the chosen formalism is based on logic. This is another proposal which does not consider the ODPs during processing. Despite ODPs, well specified, do not yield declined axioms, there is no guarantee that they will not be modified as the proposal of this dissertation can do.

As you can see, there are several proposals for implementation of ontology evolution, with different approaches, strengths and weaknesses. Therefore, some aspects were chosen for comparison: ODPs and form of implementation the process (automatic or semi-automatic). In this comparison, the ODPs indicate that the proposal had some additional concerns with the quality of the corrected ontology, while the form of implementation give us the idea of how vulnerable is the proposal regarding the inclusion of errors during the process execution. Table 5.1 contains the proposal in the order they were presented in this section.

| Proposal | ODPs | Semiautomatic Process | Automatic Process |
|---|---|---|---|
| Coulet et al [16] | No | Yes | No |
| Nikitina et al [17] | Yes | Yes | No |
| ORION | No | No | Yes |
| Rudolph et al [11] | No | No | Yes |
| Cimiano and Völker [12] | No | No | Yes |
| Corcho et al [13] | No | No | Yes |
| Friedrich and Shchekotykhin [14] | No | No | Yes |
| Friedrich and Shchekotykhin [15] | No | Yes | Yes |
| Nguyen et al [18] | No | Yes | No |
| Nikitina et al [19] | No | Yes | No |
| PRETHY | Yes | Yes | Yes |

Table 5.1: Comparison of the proposal of ontology evolution.

None of the proposals for ontology evolution presented use the same approach of the proposal of this dissertatio. In other words, none of them combine both theory revision and ODPs. Based on this recognition, no analyzes of the proposals based on the results obtained by them were made.

# 6. Conclusion

The motivation of this dissertation resides in the importance of ontology evolution stage in their life cycle. Mainly for large, complex and high quality ontologies, which have increased demand. The goal is to elaborate a ontology evolution process able to preserve the good quality of the ontology represented by ODPs and generate a coherent ontology with modeled domain. To prove the feasibility of this proposal, the following hypothesis should be confirmed:

***If*** *the process of pattern-based ontology evolution maintains the occurrences of ODPs,* ***then*** *the revision space will be reduced and a complete and correct pattern-based ontology will be obtained.*

The results point to the viability of the proposal confirming that the use of ontology design patterns is an option to ensure the quality of ontology. They are modeling solutions with recognized good quality and applicable to recurring ontology modeling problems [6, 7, 8]. They may therefore act as background knowledge in order to improve theory revision process, without preventing that coherent ontologies with the modeled domain ar obtained. The theory revision is the task oriented to transform incomplete or incorrect theories in a coherent theory from a set of examples [21].

Theory revision may be applied to automate ontology evolution. In scenarios where an ontology is constructed using ODPs, the theory revision process may take the characteristics of ODPs into account to reduce the search space of revision points, while ensuring the quality of ontology, thus improving ontology evolution process.

But for this, information that will help us during the process execution are needed. This information is organized into a data repository and allows us to identify the used ODPs. Thus, the proposal of this dissertation will be able to deal with existing ODPs in the ontology.

The proposed approach in this dissertation offers two modes (automatic and semi-automatic evaluation) to protect ODPs instances within the ontology being refined. In automatic mode, there is no possibility of errors caused by human intervention and is also the option that spend less time, but it prevents the generation of an appropriate solution in some scenarios where the errors are in ODPs. The semiautomatic mode is able to overcome the problem mentioned for the automatic option, but the disadvantage is that it is subject to errors due to human intervention, and can be time consuming.

According to the controlled experiment performed, the proposal of this dissertation performed well in most of the evaluated revision situations, especially in cases where one ODP should remain in revision space (see the last experiment of section 4.4). The PRETHY approach was able to accomplish its task in three scenarios proposed. It was able to improve the accuracy of logical theories, as well as the existing traditional process, even reducing the revision space through the protection of ODPs.

This dissertation has produced some important contributions that classify as direct and indirect. Direct contributions are those that are explicitly describe in this dissertation as the method of conversion of ODPs for FOL and the proposal of revision of pattern-based ontologies (see Chapter 3), besides examples of ODPs in FOL in A.1. As an indirect contribution, it may be mentioned that the change of the revision space causes the exploration of new solutions, which can be more efficient than the solutions found within unaltered revision space, because they are adapted to ODPs.

During the experiment of this dissertation 4, during data analysis, points of improvements that enable the development of future works were identified. Among those identified can cite the building of revision operators capable of proposing new ODPs. The reduction of the search space of revision points, in some cases of revision points generated and the number of revision operators performed, leading to a reduction in spending time also make us suggest the development of studies supporting the performance gain as future work too.

# A. Appendixes

In this appendix are described contents ontology design patterns, logical theories, facts (Bakground Knowledge) and positive and negative examples used in the experiments of chapter 4. The logical theories are in the appendix A.2, the background knowledge is in the appendix A.4 and the examples are in appendix A.5.

## A.1 Content Ontology Design Patterns

### A.1.1 Componency

Pattern used to represent the type relations part / whole, not transitive, without considering the temporal dimension. An object may be physical, social, mental, or substance. For example, we can consider the spark plug as part of the engine and the engine of the car. The Figure A.1 shows the diagram of this ODP.

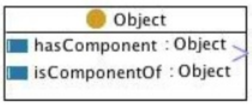


Figure A.1: Componency Pattern

Componency ODP in FOL:

object(O) :- ?(O).
isComponentOf(O2, O1) :- object(O1), object(O2), ?(O2, O1).
hasComponent(O1, O2) :- isComponentOf(O2, O1).

### A.1.2 Constituency

Pattern that allows us to represent constituents of a layered structure. It also allows us to represent physical constituents of non-physical objects, which is impossible in terms of parts. An entity is anything real, possible or imaginary we want to model. For example, a social system is characterized by layers formed by different people. The Figure A.2 shows the diagram of this ODP.

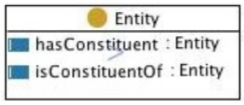


Figure A.2: Constituency Pattern

Constituency ODP in FOL:

entity(E) :- ?(E).
isConstituentOf(E2, E1) :- entity(E1), entity(E2), ?(E2, E1).
hasContituent(E1, E2) :- isContituentOf(E2, E1).

### A.1.3 Classification

Pattern representing the relationships between concepts and entities. With this pattern it is possible to make statements about, for example, categories, types, roles, etc. that are considered meta-levels of an ontology. Instances of the concept represent the elements, which are placed in an ordinary domain of an ontology. A concept is a social object. And the entity is anything real, possible or imaginary we want to model. For example, Linux and Windows can be classified as operating systems. The Figure A.3 shows the diagram of this ODP.

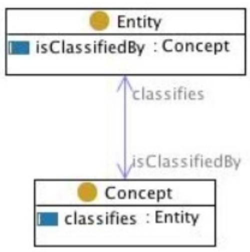


Figure A.3: Classification Pattern

Classification ODP in FOL:

entity(E) :- ?(E).

concept(C) :- ?(C).
isClassifiedBy(E, C) :- concept(C), entity(E), ?(E, C).
classifies(C, E) :- isClassifiedBy(E, C).

### A.1.4 ObjectRole

Pattern representing objects and the roles they play. With this pattern it is possible to make statements about roles, which is considered a meta-level ontology. Instances of the role represent the elements which are then placed in an ordinary domain of an ontology. A role is a concept that classifies an object. And the object can be physical, social, mental or substance. For example, use a car as a home. The Figure A.4 shows the diagram of this ODP.

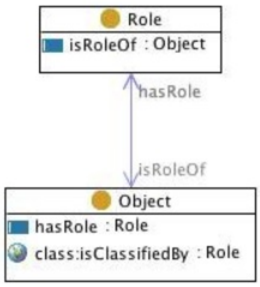


Figure A.4: ObjectRole Pattern

ObjectRole ODP in FOL:

object(O) :- ?(O).
role(R) :- ?(R).
isRoleOf(R, O) :- object(O), role(R), ?(R, O).
hasRole(O, R) :- isRoleOf(R, O).
isClassifiedBy(O, R) :- hasRole(O, R).

### A.1.5 AgentRole

Pattern representing the agents and the roles they represent. It allows us to make statements about the roles played by the agent without involving him. The agent is any executor object of action, whether physical or social. A role is a concept that classifies the object. For example, a person who plays the roles of mother, daughter and sister. The Figure A.5 shows the diagram of this ODP.

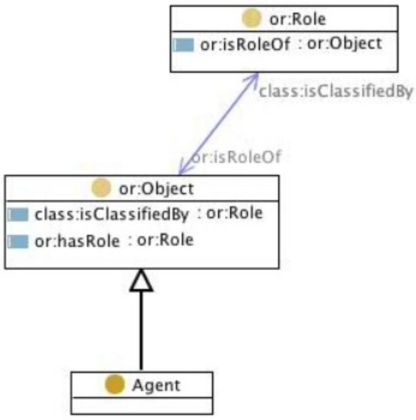


Figure A.5: AgentRole Pattern

AgentRole ODP in FOL:

object(O) :- ?(O).
agent(A) :- object(A), ?(A).
role(R) :- ?(R).
isRoleOf(R, A) :- agent(A), role(R), ?(R, A).
isClassifiedBy(A, R) :- isRoleOf(R, A).
hasRole(A, R) :- isClassifiedBy(A, R).

### A.1.6 Parameter

Pattern representing the parameters that are used by certain concepts. A parameter is a concept that restricts the attributes that an entity classified by the concept, may have in a given situation. A concept is a social object defined by a description. For example, the conditions set by the law of clean candidates to elective positions. The Figure A.6 shows the diagram of this ODP.

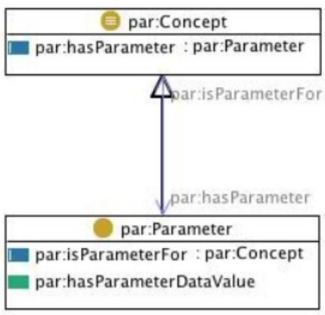


Figure A.6: Parameter Pattern

Parameter ODP in FOL:

concept(C) :- ?(C).
parameter(P) :- concept(P), ?(P).
parameterDataValue(V) :- ?(V).
isParameterDataValue(V, P) :- parameter(P), parameterDataValue(V), ?(V, P).
isParameterFor(P, C) : concept(C), parameter(P), ?(P, C).
hasParameter(C, P) :- isParameterFor(P, C).
hasParameterDataValue(P, V) :- isParameterDataValue(V, P).

### A.1.7 Description

Pattern used to formally represent a concept or a descriptive context. This pattern allows us to represent both the context (descriptive) and the elements that characterize and are involved. The description is a conceptualization. And the concept may be an idea, role or a class representation, which is defined in the description. For example, a plan is a description of some actions to be executed by agents in a certain way with certain parameters. The Figure A.7 shows the diagram of this ODP.

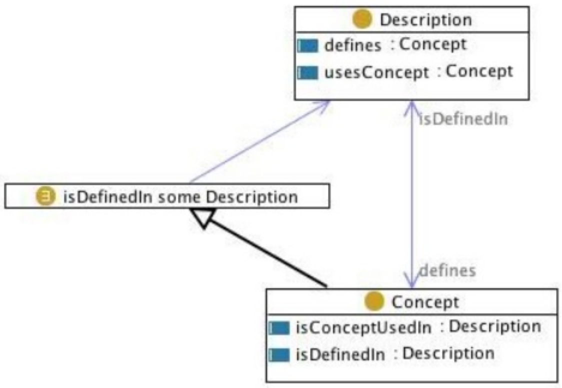


Figure A.7: Description Pattern

Description ODP in FOL:

description(D) :- ?(D).
concept(C) :- ?(C).
isConceptUsedIn(C, D) :- description(D), concept(C), ?(C, D).
isDefinedIn(C, D) :- description(D), concept(C), ?(C, D).
usesConcept(D, C) :- isConceptUsedIn(C, D).
defines(D, C) :- isDefinedIn(C, D).

### A.1.8 Observation

Pattern representing the observations of things, under a set of parameters. This pattern allows us to represent the parameters of the observations made. An observation is a specific situation where something is observed for a set of parameters. And the parameters of an observation describe the context and content of the observation. For example, we mention in the medical domain, observation of a patient may contain a set of symptoms that together are the parameters of the observation. The Figure A.8 shows the diagram of this ODP.

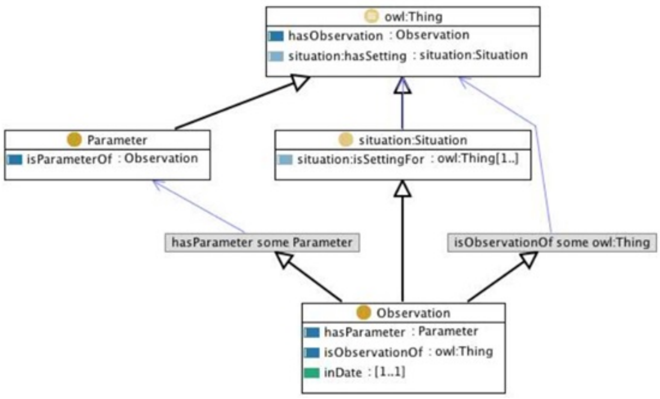


Figure A.8: Observation Pattern

Observation ODP in FOL:

thing(T) :- ?(T).
parameter(P) :- thing(P), ?(P).
situation(S) :- thing(S), ?(S).
observation(O) :- situation(S), ?(O).
date(D) :- ?(D).
isParameterOf(P, O) :- observation(O), parameter(P), ?(P, O).
observationInDate(O, D) :- observation(O), date(D), ?(O, D).
isObservationOf(O, T) :- thing(T), observation(O), ?(O, T).
isSettingFor(S, T) :- thing(T), situation(S), ?(S, T).
hasParameter(O, P) :- isParameterOf(P, O).
inDate(O, D) :- observationInDate(O, D).
hasObservation(T, O) :- isObservationOf(O, T).
hasSetting(T, S) :- isSettingFor(S, T).

### A.1.9 ObjectWithState

Pattern that allows the representation of the different states of an object and the constraints associated with each state. The states must be modeled as individuals and not as literal. An object is an entity that has different states and in each state different restrictions on their properties. And the states are the different states that an object can have. For example, in a system of control orders, an order can only be delivered when ready. The Figure A.9 shows the diagram of this ODP.

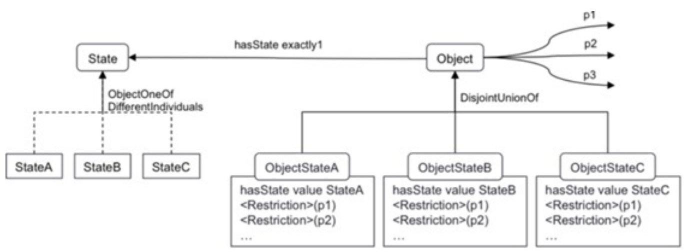


Figure A.9: ObjectWithState Pattern

ObjectWithState ODP in FOL:

object(O) :- ?(O).
state(S) :- ?(O).
restriction(R) :- ?(R).
isStateOf(S, O) :- object(O), state(S), ?(S, O).
isRestrictionOf(R, S) :- state(S), restriction(R),?(R, S).
hasState(O, S) :- isStateOf(S, O).
hasRestriction(S, R) :- isRestrictionOf(R, S).

### A.1.10 Criterion

The purpose of this pattern is to enable the modeling criteria. A criterion is a subconcept of description. For example, it allows us to model the criteria for approval in a contest. The Figure A.10 shows the diagram of this ODP.

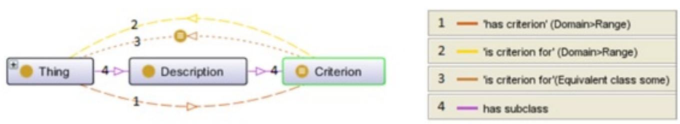


Figure A.10: Criterion Pattern

Criterion ODP in FOL:

thing(T) :- ?(T).
description(D) :- thing(D), ?(D).
criterion(C) :- description(C), ?(C).
isCriterionFor(C, T) :- thing(T), criterion(C), ?(C, T).
hasCriterion(T, C) :- isCriterionFor(C, T).

### A.1.11 CriterionSetter

The purpose of this pattern is to extend the modeling criteria. Requirements, recommendations and restrictions are possible assemblers criteria. The Figure A.11 shows the diagram of this ODP.

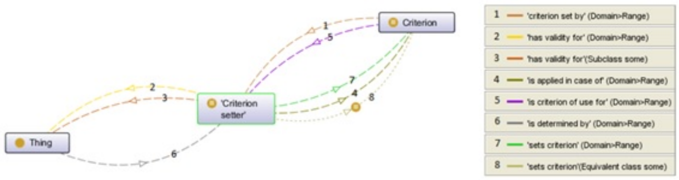


Figure A.11: CriterionSetter Pattern

CriterionSetter ODP in FOL:

thing(T) :- ?(T).
criterion(C) :- ?(C).
criterionSetter(CS) :- ?(CS).
criterionSetBy(C, CS) :- criterion(C), criterionSetter(CS), ?(C, CS).
isCriterionOfUseFor(C, CS) :- criterion(C), criterionSetter(CS), ?(C, CS).
isDeterminedBy(T, CS) :- criterionSetter(CS), thing(T), ?(T, CS).
setsCriterion(CS, C) :- criterionSetBy(C, CS).
isAppliedInCaseOf(CS, C) :- isCriterionOfUseFor(C, CS).
hasValidityFor(CS, T) :- isDeterminedBy(T, CS).

### A.1.12 Nary Participation

Pattern for representation of events with its participants, times and locations. Allows us to represent any type of relationship of this nature and the change of one of its dimensions, a new instance of the pattern must be created. A situation is an insight into a set of entities. Can be seen as a ”relational context” that represents a relation. NAryParticipation is a sub-concept of the situation representing the relation of the same name. A time interval is any region of a two dimensional space that represents time. An event is any physical, social or mental event or state process. An object is any physical, social or mental, substance or object. For example, the participation of a speaker at a particular audience at a particular time at a graduation event. The Figure A.12 shows the diagram of this ODP.

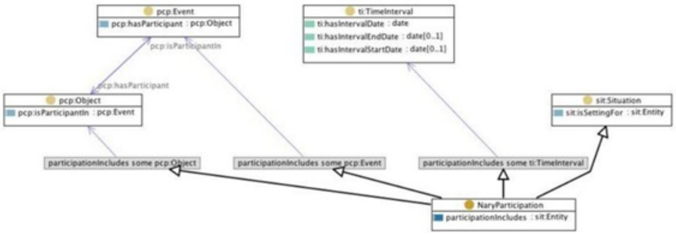


Figure A.12: Nary Participation Pattern

Nary Participation ODP in FOL:

situation(S) :- ?(S).
timeInterval(T) :- ?(T).
nAryParticipation(NAP) :- situation(NAP), ?(NAP).
object(O) :- ?(O).
event(E) :- ?(E).
hasDate(T, D) :- ?(T, D).
hasStart(T, S) :- ?(T, S).
hasFinish(T, F) :- ?(T, F).
isParticipantIn(O, E) :- event(E), object(O), ?(O, E).
isTimeInterval(T) :- timeInterval(T), hasDate(T, D), hasStart(T, S), hasFinish(T, F).
hasParticipant(E, O) :- isParticipantIn(O, E).
isIncludedInParticipation(NAP, E, O, T) :- nAryParticipation(NAP), event(E), object(O), isTimeInterval(T), hasParticipant(E, O), ?(NAP, E, O, T).
participationIncludes(NAP, E, O, T) :- isIncludedInParticipation(NAP, E, O, T).

### A.1.13 ParticipantRole

This pattern aims to represent participants and their role in a particular event. It ignores any time-related participation aspect. An event is any physical, social or mental event or state process. An object is any physical, social or mental, substance or object. A role is a concept that classifies the object. And a ParticipantRole concept is a situation that represents the role of a specific object participating in an event. For example, the appointment of an employee to occupy a leading position during the board meeting of the company. The Figure A.13 shows the diagram of this ODP.

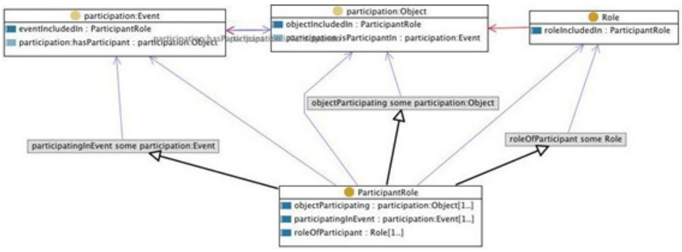

Figure A.13: ParticipantRole Pattern

ParticipantRole ODP in FOL:

object(O) :- ?(O).
event(E) :- ?(E).
role(R) :- ?(R).
participantRole(PR) :- ?(PR).
isParticipantIn(O, E) :- object(O), event(E), ?(O, E).
isRoleOf(R, O) :- role(R), object(O), ?(R, O).
hasParticipant(E, O) :- isParticipantIn(O, E).
eventIncludedIn(E, PR) :- event(E), participantRole(PR), ?(E, PR).
objectIncludedIn(O, PR) :- object(O), participantRole(PR), ?(O, PR).
roleIncludedIn(R, PR) :- role(R), participantRole(PR), ?(R, PR).
participatingInEvent(PR, E) :- eventIncludedIn(E, PR).
objectParticipanting(O, PR) :- objectIncludeIn(O, PR).
roleOfParticipant(R, O, PR) :- roleIncludedIn(R, PR), isRoleOf(R, O).
participatingIn(E, O, R, PR) :- participantingInEvent(PR, E), objectParticipanting(O, PR), roleOfParticipant(R, O, PR), hasParticipant(E, O).

### A.1.14 PartOf

Pattern for representing entities and their parts (eg, part / whole relationships) with transitivity. It is not possible to express temporal aspects of these relationships using this pattern. An entity is anything real, possible or imaginary we want to model. For example, the relationship between the human body and its various organs. The Figure A.14 shows the diagram of this ODP.

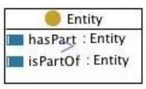


Figure A.14: PartOf Pattern

PartOf ODP in FOL:

entity(E) :- ?(E).
isPartOf(E2, E1) :- entity(E1), entity(E2), ?(E2, E1).
hasPart(PartOf, P) :- member(P, PartOf).
hasPart(E1, E2) :- bagof(E, isPartOf(E, E1), PartOf), hasPart(PartOf, E2).

### A.1.15 Bag

This pattern aims to model the bags. A bag is characterized by a collection may have multiple copies of each object. An item is an element belonging to the bag. A collection represents the concept of a set of entities (things) that share one or more properties. For example, the collection of videos from a video store can be considered a bag. The Figure A.15 shows the diagram of this ODP.

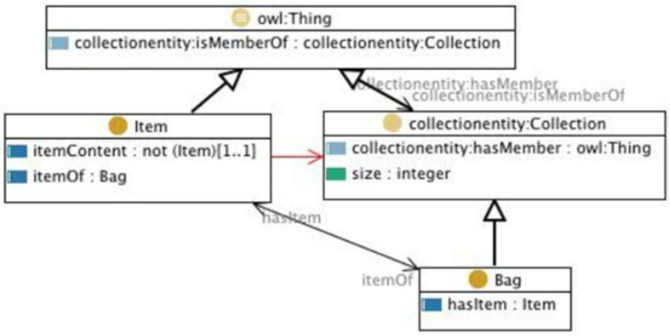


Figure A.15: Bag Pattern

Bag ODP in FOL:

thing(T) :- ?(T).
item(I) :- thing(I), ?(I).
collection(C) :- thing(C), ?(C).
bag(B) :- collection(C), ?(B).
itemOf(I, B) :- item(I), bag(B), bagof(X, ?(X, B), Bag), hasItem(Bag, I).
hasItem(B, I) :- itemOf(I, B).
hasItem(Bag, I) :- member(I, Bag).
size([],0).
size([H | T], S) :- size(T, S2), S is S2+1.
size(B, S) :- bag(B), bagof(X, itemOf(X, B), Bag), size(Bag, S).

### A.1.16 List

This pattern is used to represent ordered collections (lists) through specialization pattern bag. A ListItem concept is an element that belongs to a list. A list is an ordered array of items that can be duplicated. A bag is characterized by a collection may have multiple copies of each object. A collection represents the concept of a set of entities (things) that share one or more properties. For example, the starting grid of a race car can be represented this way. The Figure A.16 shows the diagram of this ODP.

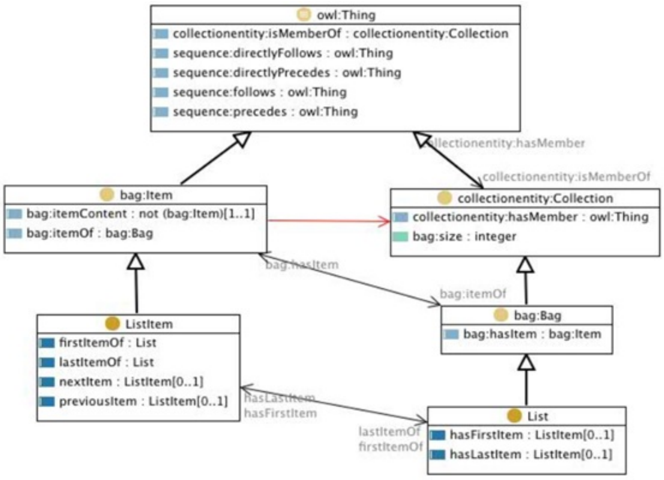


Figure A.16: List Pattern

List ODP in FOL:

```
thing(T) :- ?(T).
bagItem(BI) :- thing(BI), ?(BI).
collection(C) :- thing(C), ?(CE).
listItem(LI) :- bagItem(LI), ?(LI).
bag(B) :- collection(B), ?(B).
list([H | T]).
list([]).
list(L, List) :- bag(L), ?(L, List), list(List).
isFirstItem(H, [H | T]).
isLastItem(H, [H | []].
isLastItem(I, [H | T]) :- isLastItem(I, T).
itemOf(I, L) :- listItem(I), list(L, List), member(I, List).
hasItem(L, I) :- itemOf(I, L).
firstItemOf(I, L) :- listItem(I), list(L, List), isFirstItem(I, List).
lastItemOf(I, L) :- listItem(I), list(L, List), isLastItem(I, List).
size([],0).
size([H | T], S) :- size(T, S2), S is S2+1.
```

### A.1.17 Set

Pattern of representation of sets of things. A set is a collection that can not contain duplicate elements. A collection represents the concept of a set of entities (things) that share one or more properties. For example, the set of cards in a deck. The Figure A.17 shows the diagram of this ODP.

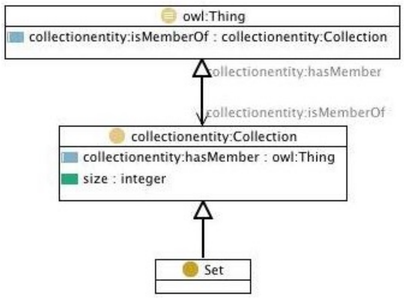


Figure A.17: Set Pattern

Set ODP in FOL:

thing(M) :- ?(M).
collectionEntity(CE) :- thing(CE), ?(CE).
set(S) :- collectionEntity(S), ?(S).
hasMember(Set, M) :- member(M, Set).
isMemberOf(M, S) :- thing(M), set(S), setof(X, ?(X, S), Set), hasMember(Set, M).
hasMember(S, M) :- isMemberOf(M, S).
size([],0).
size([H | T], S) :- size(T, S2), S is S2+1.
size(Id, S) :- set(Id), setof(X, isMemberOf(X, Id), Set), size(Set, S).

### A.1.18 Situation

This pattern represents contexts or situations and things that are contextualized. So things with something in common can be contextualised or associated. The situation is a view of a set of entities and may be seen as a relational context that represents a relation. For example, the n-ary relationship. The Figure A.18 shows the diagram of this ODP.

Situation ODP in FOL:

thing(T) :- ?(T).
situation(S) :- ?(S).

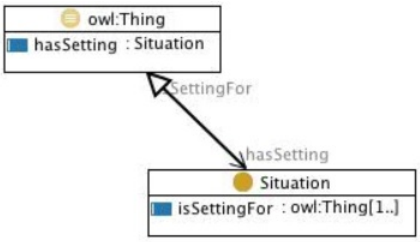


Figure A.18: Situation Pattern

isSettingFor(S, T) :- thing(T), situation(S), ?(S, T).
hasSetting(T, S) :- isSettingFor(S, T).

### A.1.19 TimeInterval

Pattern used to represent the time interval. The dates of the time interval are not part of the domain. If the need for calculation of dates, this pattern should be used along with the pattern Region. A time interval is any region in a two dimensional space that represents time. For example, the time interval January 2013 starts on 01/01/2013 and ends on 31/01/2013. The Figure A.19 shows the diagram of this ODP.

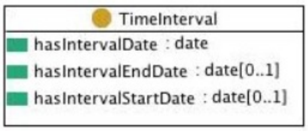


Figure A.19: TimeInterval Pattern

TimeInterval ODP in FOL:

timeInterval(T) :- ?(T).
hasDate(T, D) :- ?(T, D).
hasStart(T, S) :- ?(T, S).
hasFinish(T, F) :- ?(T, F).
isTimeInterval(T) :- timeInterval(T), hasDate(T, D), hasStart(T, S), hasFinish(T, F).

### A.1.20 Region

Pattern used to represent and reason about the values of the attributes of things, to explicitly tell the dimensions (regions) with the values of the attributes involved. A region is a dimensional space that can be used as values for one of an entity. An entity is anything real, possible or imaginary to model. For example, in the sentence ”The color of my car is red", the car is an entity and the color is the region mentioned. The color of the car (red) is one of the possible values that the region (color) defines. The Figure A.20 shows the diagram of this ODP.

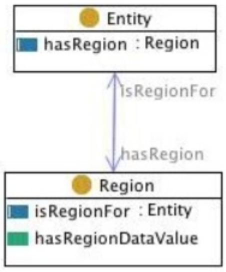


Figure A.20: Region Pattern

Region ODP in FOL:

region(R) :- ?(R).
entity(E) :- ?(E).
regionDataValue(V) :- ?(V).
isRegionFor(R, E) :- entity(E), region(R), ?(R, E).
isRegionDataValue(V, R) :- regionDataValue(V), region(R), ?(V, R).
hasRegion(E, R) :- isRegionFor(R, E).
hasRegionDataValue(R, V) :- isRegionDataValue(V, R).

## A.2 Logical Theories

In this appendix, three patterns-based ontologies (logical theories) the domain of family law are presented. The first was built based on the Brazilian Civil Code and must go through a revision process (generalization) that makes it coherent with the decision of the Brazilian Supreme Court, needing that the non patterned clauses (41) and (42) are revised for this. The second is an incomplete theory and was built based on decision of the Brazilian Supreme Court and must go through a revision process (specialization) that makes it coherent with the Brazilian Civil Code. The third was built based on the Brazilian Civil Code and must go through a revision process (generalization) that makes it coherent with the decision of the Brazilian Supreme Court, needing that the patterned clause (7) is revised for this.

### A.2.1 First Logical Theory

```
member(M) :- magistrate(M).
community(C) :- group(C).
isMemberOf(M, C) :- member(M), community(C), belongs(M, C).
hasMember(C, M) :- isMemberOf(M, C).
event(E) :- marriage(E).
object(O) :- person(O).
isParticipantIn(O, E) :- object(O), event(E), participatesIn(O, E).
hasCoParticipant(E, O) :- isParticipantIn(O, E).
coParticipatesWith(O1, O2, E) :- isParticipantIn(O1, E), isParticipantIn(O2, E).
siblings(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(Z, Y), X \== Y.
uncles(X, Y) :- siblings(X, Z), fdt:progenitor(Z, Y).
uncles(X, Y) :- fdt:married(X, Z), siblings(Z, K), progenitor(K, Y).
cousins(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(K, Y), siblings(Z, K).
grandParents(X, Y) :- fdt:progenitor(X, Z), fdt:progenitor(Z, Y).
grandFather(X, Y) :- fdt:gender(X, male), grandParents(X, Y).
grandMother(X, Y) :- fdt:gender(X, female), grandParents(X, Y).
father(X, Y) :- fdt:gender(x, male), fdt:progenitor(X, Y).
mother(X, Y) :- fdt:gender(X, female), fdt:progenitor(X, Y).
son(X, Y) :- fdt:gender(X, male), fdt:progenitor(Y, X).
daughter(X, Y) :- fdt:gender(X, female), fdt:progenitor(Y, X).
brother(X, Y) :- fdt:gender(X, male), siblings(X, Y).
sister(X, Y) :- fdt:gender(X, female), siblings(X, Y).
uncle(X, Y) :- fdt:gender(X, male), uncles(X, Y).
aunt(X, Y) :- fdt:gender(X, female), uncles(X, Y).
nephew(X, Y) :- fdt:gender(X, male), uncles(Y, X).
niece(X, Y) :- fdt:gender(X, female), uncles(Y, X).
cousin(X, Y) :- fdt:gender(X, male), cousins(X, Y).
press(X, Y) :- fdt:gender(X, female), cousins(X, Y).
ancestral(X, Y) :- fdt:progenitor(X, Y).
ancestral(X, Y) :- fdt:progenitor(Z, Y), ancestral(X, Z).
descendent(X, Y) :- fdt:progenitor(Y, X).
descendent(X, Y) :- fdt:progenitor(Y, Z), descendent(X, Z).
isMarried(X) :- fdt:married(X, Y), not fdt:dead(Y).
civilSituationOk(H, W) :- fdt:single(H), fdt:single(W).
ageSituationOk(H, W) :- fdt:adult(H), fdt:adult(W).
```

36 ageSituationOk(H, W) :- fdt:adult(H), fdt:adolescent(W), fdt:havePermissionToMarry(W).
37 ageSituationOk(H, W) :- fdt:adolescent(H), fdt:havePermissionToMarry(H), fdt:adult(W).
38 ableToCivilMarriage(H, W) :- ageSituationOk(H, W), civilSituationOk(H, W), H \== W.
39 allowedMarriage(J, E) :- fdt:isParticipantIn(J, E), fdt:isMemberOf(J, tj).
40 canMarry(H, W, E) :- fdt:coParticipatesWith(H, W, E), ableToCivilMarriage(H, W).
41 civilMarriage(H, W, J, E) :- gender(H, male), gender(W, female), allowedMarriage(J, E), canMarry(H, W, E).
42 civilMarriage(H, W, J, E) :- gender(H, female), gender(W, male), allowedMarriage(J, E), canMarry(H, W, E).

Table A.1: Logical theory used in a generalization process

### A.2.2 Second Logical Theory

1 member(M) :- magistrate(M).
2 community(C) :- group(C).
3 isMemberOf(M, C) :- member(M), community(C), belongs(M, C).
4 hasMember(C, M) :- isMemberOf(M, C).
5 event(E) :- marriage(E).
6 object(O) :- person(O).
7 isParticipantIn(O, E) :- object(O), event(E), participatesIn(O, E).
8 hasCoParticipant(E, O) :- isParticipantIn(O, E).
9 coParticipatesWith(O1, O2, E) :- isParticipantIn(O1, E), isParticipantIn(O2, E).
10 siblings(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(Z, Y), X \== Y.
11 uncles(X, Y) :- siblings(X, Z), fdt:progenitor(Z, Y).
12 uncles(X, Y) :- fdt:married(X, Z), siblings(Z, K), progenitor(K, Y).
13 cousins(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(K, Y), siblings(Z, K).
14 grandParents(X, Y) :- fdt:progenitor(X, Z), fdt:progenitor(Z, Y).
15 grandFather(X, Y) :- fdt:gender(X, male), grandParents(X, Y).
16 grandMother(X, Y) :- fdt:gender(X, female), grandParents(X, Y).
17 father(X, Y) :- fdt:gender(x, male), fdt:progenitor(X, Y).
18 mother(X, Y) :- fdt:gender(X, female), fdt:progenitor(X, Y).
19 son(X, Y) :- fdt:gender(X, male), fdt:progenitor(Y, X).
20 daughter(X, Y) :- fdt:gender(X, female), fdt:progenitor(Y, X).
21 brother(X, Y) :- fdt:gender(X, male), siblings(X, Y).
22 sister(X, Y) :- fdt:gender(X, female), siblings(X, Y).

23 uncle(X, Y) :- fdt:gender(X, male), uncles(X, Y).
24 aunt(X, Y) :- fdt:gender(X, female), uncles(X, Y).
25 nephew(X, Y) :- fdt:gender(X, male), uncles(Y, X).
26 niece(X, Y) :- fdt:gender(X, female), uncles(Y, X).
27 cousin(X, Y) :- fdt:gender(X, male), cousins(X, Y).
28 press(X, Y) :- fdt:gender(X, female), cousins(X, Y).
29 ancestral(X, Y) :- fdt:progenitor(X, Y).
30 ancestral(X, Y) :- fdt:progenitor(Z, Y), ancestral(X, Z).
31 descendent(X, Y) :- fdt:progenitor(Y, X).
32 descendent(X, Y) :- fdt:progenitor(Y, Z), descendent(X, Z).
33 isMarried(X) :- fdt:married(X, Y), not fdt:dead(Y).
34 civilSituationOk(H, W) :- fdt:single(H), fdt:single(W).
35 ageSituationOk(H, W) :- fdt:adult(H), fdt:adult(W).
36 ageSituationOk(H, W) :- fdt:adult(H), fdt:adolescent(W), fdt:havePermissionToMarry(W).
37 ageSituationOk(H, W) :- fdt:adolescent(H), fdt:havePermissionToMarry(H), fdt:adult(W).
38 ableToCivilMarriage(H, W) :- ageSituationOk(H, W), civilSituationOk(H, W), H \== W.
39 allowedMarriage(J, E) :- fdt:isParticipantIn(J, E), fdt:isMemberOf(J, tj).
40 canMarry(H, W, E) :- fdt:coParticipatesWith(H, W, E), ableToCivilMarriage(H, W).
41 civilMarriage(H, W, J, E) :- allowedMarriage(J, E), canMarry(H, W, E).

Table A.2: Logical theory used in a specialization process

### A.2.3 Third Logical Theory

1 member(M) :- fdt:magistrate(M).
2 community(C) :- fdt:group(C).
3 isMemberOf(M, C) :- community(C), member(M), fdt:belongs(M, C).
4 hasMember(C, M) :- isMemberOf(M, C).
5 object(O) :- fdt:person(O).
6 event(E) :- fdt:matrimony(E).
7 isParticipantIn(O, E) :- object(O), event(E), participatesIn(O, E), isInvited(O, E).
8 hasParticipant(E, O) :- isParticipantIn(O, E).
9 coParticipatesWith(O1, O2, E) :- isParticipantIn(O1, E), isParticipantIn(O2, E).
10 siblings(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(Z, Y), X \== Y.
11 uncles(X, Y) :- siblings(X, Z), fdt:progenitor(Z, Y).
12 uncles(X, Y) :- fdt:married(X, Z), siblings(Z, K), fdt:progenitor(K, Y).
13 cousins(X, Y) :- fdt:progenitor(Z, X), fdt:progenitor(K, Y), siblings(Z, K).
14 grandParents(X, Y) :- fdt:progenitor(X, Z), fdt:progenitor(Z, Y).

```
15 grandFather(X, Y) :- fdt:gender(X, male), grandParents(X, Y).
16 grandMother(X, Y) :- fdt:gender(X, female), grandParents(X, Y).
17 father(X, Y) :- fdt:gender(x, male), fdt:progenitor(X, Y).
18 mother(X, Y) :- fdt:gender(X, female), fdt:progenitor(X, Y).
19 son(X, Y) :- fdt:gender(X, male), fdt:progenitor(Y, X).
20 daughter(X, Y) :- fdt:gender(X, female), fdt:progenitor(Y, X).
21 brother(X, Y) :- fdt:gender(X, male), siblings(X, Y).
22 sister(X, Y) :- fdt:gender(X, female), siblings(X, Y).
23 uncle(X, Y) :- fdt:gender(X, male), uncles(X, Y).
24 aunt(X, Y) :- fdt:gender(X, female), uncles(X, Y).
25 nephew(X, Y) :- fdt:gender(X, male), uncles(Y, X).
26 niece(X, Y) :- fdt:gender(X, female), uncles(Y, X).
27 cousin(X, Y) :- fdt:gender(X, male), cousins(X, Y).
28 press(X, Y) :- fdt:gender(X, female), cousins(X, Y).
29 ancestral(X, Y) :- fdt:progenitor(X, Y).
30 ancestral(X, Y) :- fdt:progenitor(Z, Y), ancestral(X, Z).
31 descendent(X, Y) :- fdt:progenitor(Y, X).
32 descendent(X, Y) :- fdt:progenitor(Y, Z), descendent(X, Z).
33 civilSituationOk(H, W) :- fdt:single(H), fdt:single(W).
34 ageSituationOk(H, W) :- fdt:adult(H), fdt:adult(W).
35 ageSituationOk(H, W) :- fdt:adult(H), fdt:adolescent(W), fdt:havePermissionToMarry(W).
36 ageSituationOk(H, W) :- fdt:adolescent(H), fdt:havePermissionToMarry(H), fdt:adult(W) .
37 ableToCivilMarriage(H, W) :- ageSituationOk(H, W), civilSituationOk(H, W), H \== W.
38 allowedMarriage(J, E) :- isParticipantIn(J, E), isMemberOf(J, tj).
39 canMarry(H, W, E) :- coParticipatesWith(H, W, E), ableToCivilMarriage(H, W).
40 civilMarriage(H, W, J, E) :- canMarry(H, W, E), allowedMarriage(J, E).
```

Table A.3: Logical theory used in a generalization process

### A.3 Proposed Revisions

#### A.3.1 Proposed Revisions by FORTE System with $R = R'$

1ª Proposed revision

civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).

civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).

2ª Proposed revision

civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(B,A,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

3ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

4ª Proposed revision
civilMarriage(A,B,C,D):-gender(B,male).
civilMarriage(A,B,C,D):-canMarry(B,A,D),allowedMarriage(C,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

5ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

6ª Proposed revision
civilSituationOk(A,B):-gender(B,male).
civilSituationOk(A,B):-fdt:single(A),fdt:single(B).

7ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

8ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

9ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).

civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

10ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

11ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

12ª Proposed revision
allowedMarriage(A,B):-isMemberOf(A,tj),hasParticipant(B,A).

13ª Proposed revision
allowedMarriage(A,B):-isParticipantIn(A,B),hasMember(tj,A).

Table A.4: Proposed revisions by FORTE with $R = R'$

### A.3.2 Proposed Revisions by PRETHY in Automatic Mode with $R = R'$

1ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).

2ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(B,A,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

3ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

4ª Proposed revision
civilMarriage(A,B,C,D):-gender(B,male).

civilMarriage(A,B,C,D):-canMarry(B,A,D),allowedMarriage(C,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

5ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

6ª Proposed revision
civilSituationOk(A,B):-gender(B,male).
civilSituationOk(A,B):-fdt:single(A),fdt:single(B).

7ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

8ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

9ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),canMarry(A,B,D).

10ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

11ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

Table A.5: Proposed revisions by Automatic mode $R = R'$

### A.3.3 Proposed Revisions by PRETHY in Semiautomatic Mode with $R = R'$

1ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),
canMarry(A,B,D).

2ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(B,A,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),
canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),
canMarry(A,B,D).

3ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),
canMarry(A,B,D).

4ª Proposed revision
civilMarriage(A,B,C,D):-gender(B,male).
civilMarriage(A,B,C,D):-canMarry(B,A,D),allowedMarriage(C,D).
civilMarriage(A,B,C,D):-gender(A,female),gender(B,male),allowedMarriage(C,D),
canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),
canMarry(A,B,D).

5ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

6ª Proposed revision
civilSituationOk(A,B):-gender(B,male).

civilSituationOk(A,B):-fdt:single(A),fdt:single(B).

7ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

8ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),
canMarry(A,B,D).

9ª Proposed revision
civilMarriage(A,B,C,D):-allowedMarriage(C,D),gender(B,male).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).
civilMarriage(A,B,C,D):-gender(A,male),gender(B,female),allowedMarriage(C,D),
canMarry(A,B,D).

10ª Proposed revision
canMarry(A,B,C):-gender(B,male).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B).

11ª Proposed revision
ableToCivilMarriage(A,B):-gender(B,male).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B.

12ª Proposed revision
allowedMarriage(A,B):-isMemberOf(A,tj),hasParticipant(B,A).

Table A.6: Proposed revisions by Semiautomatic mode $R = R'$

### A.3.4 Proposed Revisions by FORTE System in the Specialization Process

---

1ª Proposed revision
civilMarriage(A,B,C,D):-fail.

2ª Proposed revision
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(B,male), gender(A,female).
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(A,male), gender(B,female).

3ª Proposed revision
coParticipatesWith(A,B,C):-fail.

4ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C), gender(B,male),gender(A,female).
coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C), gender(A,male),gender(B,female).

5ª Proposed revision
canMarry(A,B,C):-fail.

6ª Proposed revision
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B), gender(B,male),gender(A,female).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B), gender(A,male),gender(B,female).

7ª Proposed revision
isParticipantIn(A,B):-fail.

8ª Proposed revision
isMemberOf(A,B):-fail.

9ª Proposed revision
isMemberOf(A,B):-community(B),member(A),fdt:belongs(A,B),gender(A,male).

10ª Proposed revision
civilSituationOk(A,B):-fail.

10ª Proposed revision
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(B,male),gender(A,female).
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(A,male),gender(B,female).

11ª Proposed revision
allowedMarriage(A,B):-fail.

12ª Proposed revision
allowedMarriage(A,B):-isParticipantIn(A,B),isMemberOf(A,tj),magistracy(A,chiefJudge).

13ª Proposed revision
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

14ª Proposed revision
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(B,male),gender(A,female).
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(A,male),gender(B,female).
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

15ª Proposed revision
ableToCivilMarriage(A,B):-fail.

16ª Proposed revision
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,gender(B,male), gender(A,female).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,gender(A,male), gender(B,female).

17ª Proposed revision
object(A):-fail.

18ª Proposed revision

member(A):-fail.

19ª Proposed revision
member(A):-fdt:magistrate(A),gender(A,male).
member(A):-fdt:magistrate(A),magistracy(A,chiefJudge),gender(A,male).

20ª Proposed revision
event(A):-fail.

21ª Proposed revision
community(A):-fail.

22ª Proposed revision
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(B,male),gender(A,female).
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(A,male),gender(B,female).

23ª Proposed revision
allowedMarriage(A,B):-isMemberOf(A,tj),hasParticipant(B,A).

24ª Proposed revision
allowedMarriage(A,B):-isParticipantIn(A,B),hasMember(tj,A).

Table A.7: Proposed revisions by FORTE (specialization process)

### A.3.5 Proposed Revisions by PRETHY in Automatic Mode in the Specialization Process

1ª Proposed revision
civilMarriage(A,B,C,D):-fail.

2ª Proposed revision
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(B,male),gender(A,female).
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(A,male),gender(B,female).

3ª Proposed revision
canMarry(A,B,C):-fail.

4ª Proposed revision
canMarry(A,B,C):-fdt:coParticipatesWith(A,B,C),ableToCivilMarriage(A,B),gender(B,male), gender(A,female).
canMarry(A,B,C):-fdt:coParticipatesWith(A,B,C),ableToCivilMarriage(A,B),gender(A,male), gender(B,female).

5ª Proposed revision
civilSituationOk(A,B):-fail.

6ª Proposed revision
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(B,male),gender(A,female).
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(A,male),gender(B,female).

7ª Proposed revision
allowedMarriage(A,B):-fail.

8ª Proposed revision
allowedMarriage(A,B):-fdt:isParticipantIn(A,B),fdt:isMemberOf(A,tj),magistracy(A,chiefJudge).

9ª Proposed revision
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

10ª Proposed revision
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(B,male),gender(A,female).
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(A,male),gender(B,female).
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

11ª Proposed revision
ableToCivilMarriage(A,B):-fail.

12ª Proposed revision
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,gender(B,male), gender(A,female).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,gender(A,male), gender(B,female).

Table A.8: Proposed revisions by Automatic mode (Specialization Process)

### A.3.6 Proposed Revisions by PRETHY in Semiautomatic Mode in the Specialization Process

1ª Proposed revision
civilMarriage(A,B,C,D):-fail.

2ª Proposed revision
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(B,male), gender(A,female).
civilMarriage(A,B,C,D):-canMarry(A,B,D),allowedMarriage(C,D),gender(A,male), gender(B,female).

3ª Proposed revision
coParticipatesWith(A,B,C):-fail.

4ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C),gender(B,male), gender(A,female).
coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C),gender(A,male), gender(B,female).

5ª Proposed revision
canMarry(A,B,C):-fail.

6ª Proposed revision
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B),gender(B,male), gender(A,female).
canMarry(A,B,C):-coParticipatesWith(A,B,C),ableToCivilMarriage(A,B),gender(A,male), gender(B,female).

7ª Proposed revision
isParticipantIn(A,B):-fail.

8ª Proposed revision
civilSituationOk(A,B):-fail.

9ª Proposed revision
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(B,male),gender(A,female).
civilSituationOk(A,B):-fdt:single(A),fdt:single(B),gender(A,male),gender(B,female).

10ª Proposed revision
allowedMarriage(A,B):-fail.

11ª Proposed revision
allowedMarriage(A,B):-isParticipantIn(A,B),fdt:isMemberOf(A,tj),magistracy(A,chiefJudge).

12ª Proposed revision
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

13ª Proposed revision
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(B,male),gender(A,female).
ageSituationOk(A,B):-fdt:adult(A),fdt:adult(B),gender(A,male),gender(B,female).
ageSituationOk(A,B):-fdt:adolescent(A),fdt:havePermissionToMarry(A),fdt:adult(B).
ageSituationOk(A,B):-fdt:adult(A),fdt:adolescent(B),fdt:havePermissionToMarry(B).

14ª Proposed revision
ableToCivilMarriage(A,B):-fail.

15ª Proposed revision
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,
gender(B,male),gender(A,female).
ableToCivilMarriage(A,B):-ageSituationOk(A,B),civilSituationOk(A,B),A\==B,
gender(A,male),gender(B,female).

16ª Proposed revision
object(A):-fail.

17ª Proposed revision
event(A):-fail.

| |
|---|
| 18ª Proposed revision<br>allowedMarriage(A,B):-fdt:isMemberOf(A,tj),hasParticipant(B,A). |

Table A.9: Proposed revisions by Semiautomatic mode (Specialization Process)

### A.3.7 Proposed Revisions by FORTE System with $R' \subset R$

| |
|---|
| 1ª Proposed revision<br>coParticipatesWith(A,B,C):-isParticipantIn(B,C),hasParticipant(C,A). |
| 2ª Proposed revision<br>coParticipatesWith(A,B,C):-isParticipantIn(A,C),hasParticipant(C,B). |
| 3ª Proposed revision<br>isParticipantIn(A,B):-participatesIn(A,B).<br>isParticipantIn(A,B):-object(A),event(B),participatesIn(A,B),isInvited(A,B). |
| 4ª Proposed revision<br>coParticipatesWith(A,B,C):-participatesIn(B,C),person(A).<br>coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C). |
| 5ª Proposed revision<br>isParticipantIn(A,B):-object(A),event(B),participatesIn(A,B). |
| 6ª Proposed revision<br>isParticipantIn(A,B):-participatesIn(A,B).<br>isParticipantIn(A,B):-object(A),event(B),participatesIn(A,B),isInvited(A,B). |
| 7ª Proposed revision<br>isParticipantIn(A,B):-object(A),event(B),participatesIn(A,B). |
| 8ª Proposed revision<br>coParticipatesWith(A,B,C):-isParticipantIn(B,C),hasParticipant(C,A). |
| 9ª Proposed revision |

coParticipatesWith(A,B,C):-isParticipantIn(A,C),hasParticipant(C,B).

Table A.10: Proposed revisions by FORTE with $R' \subset R$

### A.3.8 Proposed Revisions by PRETHY in Automatic Mode with $R' \subset R$

1ª Proposed revision
civilMarriage(A,B,C,D):-participatesIn(C,D),participatesIn(B,D),participatesIn(A,D).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).

2ª Proposed revision
civilMarriage(A,B,C,D):-participatesIn(C,D),participatesIn(B,D),participatesIn(A,D).
civilMarriage(A,B,C,D):-allowedMarriage(C,D),canMarry(A,B,D).

Table A.11: Proposed revisions by Automatic mode with $R' \subset R$

### A.3.9 Proposed Revisions by PRETHY in Semiautomatic Mode with $R' \subset R$

1ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(B,C),hasParticipant(C,A).

2ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(A,C),hasParticipant(C,B).

3ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(B,C),hasParticipant(C,A).

4ª Proposed revision
isParticipantIn(A,B):-participatesIn(A,B).
isParticipantIn(A,B):-object(A),event(B),fdt:participatesIn(A,B),fdt:isInvited(A,B).

5ª Proposed revision
coParticipatesWith(A,B,C):-participatesIn(B,C),person(A).
coParticipatesWith(A,B,C):-isParticipantIn(A,C),isParticipantIn(B,C).

6ª Proposed revision
isParticipantIn(A,B):-object(A),event(B),fdt:participatesIn(A,B).

7ª Proposed revision
isParticipantIn(A,B):-participatesIn(A,B).
isParticipantIn(A,B):-object(A),event(B),fdt:participatesIn(A,B),fdt:isInvited(A,B).

8ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(B,C),hasParticipant(C,A).

9ª Proposed revision
coParticipatesWith(A,B,C):-isParticipantIn(A,C),hasParticipant(C,B).

Table A.12: Proposed revisions by Semiautomatic mode with $R' \subset R$

## A.4 Background Knowledge

### A.4.1 Facts

person(anderson), person(solange), person(augusto), person(julieta), person(joaquim), person(maria),
person(marcelo), person(carla), person(diego), person(amanda), person(thiago), person(renata),
person(guilherme), person(lara), person(caio), person(heloisa), person(roberto), person(roberta),
person(manuel), person(eduarda), person(julio), person(lucia), person(alfredo), person(sonia),
person(jorge), person(yasmin), person(alberto), person(luiza), person(gabriel), person(sofia),
person(bruno), person(bruna), person(luiz), person(leticia), person(marcio), person(tulio),
person(helena), person(marcos), gender(anderson, male), gender(solange, female), gender(augusto, male),
gender(julieta, female), gender(joaquim, male), gender(maria, female), gender(marcelo, male),
gender(carla, female), gender(diego, male), gender(amanda, female), gender(thiago, male),
gender(renata, female), gender(guilherme, male), gender(lara, female), gender(caio, male),
gender(heloisa, female), gender(roberto, male), gender(roberta, female), gender(manuel, male),
gender(eduarda, female), gender(julio, male), gender(lucia, female), gender(alfredo, male),

gender(sonia, female), gender(jorge, male), gender(yasmin, female), gender(alberto, male), gender(luiza, female), gender(gabriel, male), gender(sofia, female), gender(bruno, male), gender(bruna, female), gender(luiz, male), gender(leticia, female), gender(marcio, male), gender(helena, female), gender(tulio, male), gender(marcos, male), magistrate(luiz), magistrate(marcio),
magistrate(leticia), magistracy(luiz, chiefJudge), magistracy(marcio, judge), magistracy(leticia, chiefJudge),
belongs(luiz, tj), belongs(marcio, tj), belongs(leticia, tj), group(tj), single(amanda), single(thiago), single(renata), single(guilherme), single(lara), single(sonia), single(alfredo), single(jorge), single(yasmin), single(alberto), single(luiza), single(sofia), single(gabriel), adult(anderson), adult(solange), adult(augusto), adult(julieta), adult(joaquim), adult(maria), adult(caio), adult(heloisa), adult(roberto), adult(roberta), adult(manuel), adult(eduarda), adult(julio), adult(lucia), adult(bruno), adult(bruna), adult(marcelo), adult(carla), adult(thiago), adult(renata), adult(sonia), adult(alfredo), adult(alberto), adult(luiza), adult(marcos), adult(tulio), adult(helena), adolescent(diego), adolescent(amanda), adolescent(guilherme),
adolescent(lara), adolescent(jorge), adolescent(yasmin), adolescent(sofia), adolescent(gabriel),
killed(tulio, helena), dead(helena), adopted(manuel, marcos), adopted(eduarda, marcos), civilKinship(anderson, julieta), civilKinship(anderson, joaquim), civilKinship(solange, julieta),
civilKinship(solange, joaquim), civilKinship(augusto, joaquim), civilKinship(maria, julieta), civilKinship(caio, maria), civilKinship(caio, augusto), civilKinship(heloisa, maria), civilKinship(heloisa, augusto), civilKinship(roberto, eduarda), civilKinship(roberto, julio), civilKinship(roberta, eduarda), civilKinship(roberta, julio), civilKinship(manuel, julio), civilKinship(lucia, eduarda), civilKinship(bruno, lucia), civilKinship(bruno, manuel), civilKinship(bruna, lucia), civilKinship(bruna, manuel), progenitor(anderson, augusto), progenitor(anderson, maria), progenitor(solange, augusto), progenitor(solange, maria), progenitor(augusto, marcelo), progenitor(augusto, carla), progenitor(augusto, diego), progenitor(augusto, amanda), progenitor(julieta, marcelo), progenitor(julieta, carla), progenitor(julieta, diego), progenitor(julieta, amanda), progenitor(joaquim, thiago), progenitor(joaquim, renata), progenitor(joaquim, guilherme), progenitor(joaquim, lara), progenitor(maria, thiago), progenitor(maria, renata), progenitor(maria, guilherme), progenitor(maria, lara), progenitor(caio, joaquim), progenitor(caio, julieta), progenitor(heloisa, joaquim), progenitor(heloisa, julieta), progenitor(roberto, manuel), progenitor(roberto, lucia), progenitor(roberta, manuel), progenitor(roberta, lucia), progenitor(manuel, sonia), progenitor(manuel, alfredo), progenitor(manuel, yasmin),

progenitor(manuel, jorge), progenitor(eduarda, sonia), progenitor(eduarda, alfredo), progenitor(eduarda, yasmin), progenitor(eduarda, jorge), progenitor(julio, alberto), progenitor(julio, luiza), progenitor(julio, gabriel), progenitor(julio, sofia), progenitor(lucia, alberto), progenitor(lucia, luiza), progenitor(lucia, gabriel), progenitor(lucia, sofia), havePermissionToMarry(diego), havePermissionToMarry(amanda), havePermissionToMarry(guilherme), havePermissionToMarry(lara), havePermissionToMarry(jorge),
havePermissionToMarry(yasmin), havePermissionToMarry(sofia), havePermissionToMarry(gabriel),
married(anderson, solange), married(augusto, julieta), married(joaquim, maria), married(caio, heloisa),
married(roberto, roberta), married(manuel, eduarda), married(julio, lucia), married(bruno, bruna),
married(marcelo, sonia), married(marcelo, alfredo), married(marcelo, jorge), married(marcelo, yasmin),
married(marcelo, alberto), married(marcelo, luiza), married(marcelo, gabriel), married(marcelo, sofia),
married(carla, sonia), married(carla, alfredo), married(carla, jorge), married(carla, yasmin), married(carla, alberto), married(carla, luiza), married(carla, gabriel), married(carla, sofia),
married(diego, sonia), married(diego, alfredo), married(diego, jorge), married(diego, yasmin),
married(diego, alberto), married(diego, luiza), married(diego, gabriel), married(diego, sofia),
married(amanda, sonia), married(amanda, alfredo), married(amanda, jorge), married(amanda, yasmin),
married(amanda, alberto), married(amanda, luiza), married(amanda, gabriel), married(amanda, sofia),
married(thiago, sonia), married(thiago, alfredo), married(thiago, jorge), married(thiago, yasmin),
married(thiago, alberto), married(thiago, luiza), married(thiago, gabriel), married(thiago, sofia),
married(renata, sonia), married(renata, alfredo), married(renata, jorge), married(renata, yasmin),
married(renata, alberto), married(renata, luiza), married(renata, gabriel), married(renata, sofia),
married(guilherme, sonia), married(guilherme, alfredo), married(guilherme, jorge),married(guilherme, yasmin),

married(guilherme, alberto), married(guilherme, luiza), married(guilherme, gabriel), married(guilherme, sofia),
married(lara, sonia), married(lara, alfredo), married(lara, jorge), married(lara, yasmin), married(lara, alberto),
married(lara, luiza), married(lara, gabriel), married(lara, sofia), matrimony(m1), matrimony(m2), matrimony(m3),
matrimony(m4), matrimony(m5), matrimony(m6), matrimony(m7), matrimony(m8), matrimony(m9), matrimony(m10),
matrimony(m11), matrimony(m12), matrimony(m13), matrimony(m14), matrimony(m15), matrimony(m16),
matrimony(m17), matrimony(m18), matrimony(m19), matrimony(m20), matrimony(m21), matrimony(m22), matrimony(m23),
matrimony(m24), matrimony(m25), matrimony(m26), matrimony(m27), matrimony(m28), matrimony(m29), matrimony(m30),
matrimony(m31), matrimony(m32), matrimony(m33), matrimony(m37), matrimony(m38), matrimony(m39),
matrimony(m40), matrimony(m41), matrimony(m42), matrimony(m43), matrimony(m44), matrimony(m45), matrimony(m46),
matrimony(m47), matrimony(m48), matrimony(m49), matrimony(m50), matrimony(m51), matrimony(m52),
matrimony(m53), matrimony(m54), matrimony(m55), matrimony(m56), matrimony(m57), matrimony(m58), matrimony(m59),
matrimony(m60), matrimony(m61), matrimony(m62), matrimony(m63), matrimony(m64), matrimony(m65), matrimony(m66),
matrimony(m67), matrimony(m68), matrimony(m69), matrimony(m70), matrimony(m71), matrimony(m72)
participatesIn(anderson, m1), participatesIn(solange, m1), participatesIn(luiz, m1), participatesIn(augusto, m1),
participatesIn(julieta, m1), participatesIn(joaquim, m1), participatesIn(maria, m1), participatesIn(caio, m1),
participatesIn(heloisa, m1), participatesIn(marcelo, m1), participatesIn(carla, m1), participatesIn(diego, m1),
participatesIn(amanda, m1), participatesIn(thiago, m1), participatesIn(renata, m1), participatesIn(guilherme, m1),
participatesIn(lara, m1), participatesIn(roberto, m1), participatesIn(roberta, m1), participatesIn(manuel, m1),

participatesIn(eduarda, m1), participatesIn(julio, m1), participatesIn(lucia, m1), participatesIn(bruno, m1),
participatesIn(bruna, m1), participatesIn(sonia, m1), participatesIn(alfredo, m1), participatesIn(jorge, m1),
participatesIn(yasmin, m1), participatesIn(alberto, m1), participatesIn(luiza, m1), participatesIn(sofia, m1),
participatesIn(gabriel, m1), participatesIn(augusto, m2), participatesIn(julieta, m2), participatesIn(marcio, m2),
participatesIn(anderson, m2), participatesIn(solange, m2), participatesIn(joaquim, m2), participatesIn(maria, m2),
participatesIn(caio, m2), participatesIn(heloisa, m2), participatesIn(marcelo, m2), participatesIn(carla, m2),
participatesIn(diego, m2), participatesIn(amanda, m2), participatesIn(thiago, m2), participatesIn(renata, m2),
participatesIn(guilherme, m2), participatesIn(lara, m2), participatesIn(roberto, m2), participatesIn(roberta, m2),
participatesIn(manuel, m2), participatesIn(eduarda, m2), participatesIn(julio, m2), participatesIn(lucia, m2),
participatesIn(bruno, m2), participatesIn(bruna, m2), participatesIn(sonia, m2), participatesIn(alfredo, m2),
participatesIn(jorge, m2), participatesIn(yasmin, m2), participatesIn(alberto, m2), participatesIn(luiza, m2),
participatesIn(sofia, m2), participatesIn(gabriel, m2), participatesIn(joaquim, m3), participatesIn(maria, m3),
participatesIn(leticia, m3), participatesIn(augusto, m3), participatesIn(julieta, m3),
participatesIn(anderson, m3), participatesIn(solange, m3), participatesIn(caio, m3), participatesIn(heloisa, m3),
participatesIn(marcelo, m3), participatesIn(carla, m3), participatesIn(diego, m3), participatesIn(amanda, m3),
participatesIn(thiago, m3), participatesIn(renata, m3), participatesIn(guilherme, m3), participatesIn(lara, m3),
participatesIn(roberto, m3), participatesIn(roberta, m3), participatesIn(manuel, m3), participatesIn(eduarda, m3),
participatesIn(julio, m3), participatesIn(lucia, m3), participatesIn(bruno, m3), participatesIn(bruna, m3),
participatesIn(sonia, m3), participatesIn(alfredo, m3), participatesIn(jorge, m3), participatesIn(yasmin, m3),

participatesIn(caio, m4), participatesIn(heloisa, m4), participatesIn(luiz, m4), participatesIn(augusto, m4),
participatesIn(julieta, m4), participatesIn(joaquim, m4), participatesIn(maria, m4),
participatesIn(anderson, m4), participatesIn(solange, m4), participatesIn(marcelo, m4), participatesIn(carla, m4),
participatesIn(diego, m4), participatesIn(amanda, m4), participatesIn(thiago, m4), participatesIn(renata, m4),
participatesIn(guilherme, m4), participatesIn(lara, m4), participatesIn(roberto, m4), participatesIn(roberta, m4),
participatesIn(manuel, m4), participatesIn(eduarda, m4), participatesIn(julio, m4), participatesIn(lucia, m4),
participatesIn(bruno, m4), participatesIn(bruna, m4), participatesIn(sonia, m4), participatesIn(alfredo, m4),
participatesIn(jorge, m4), participatesIn(yasmin, m4), participatesIn(alberto, m4), participatesIn(luiza, m4),
participatesIn(sofia, m4), participatesIn(gabriel, m4), participatesIn(roberto, m5), participatesIn(roberta, m5),
participatesIn(marcio, m5), participatesIn(augusto, m5), participatesIn(julieta, m5), participatesIn(joaquim, m5),
participatesIn(maria, m5), participatesIn(caio, m5), participatesIn(heloisa, m5), participatesIn(marcelo, m5),
participatesIn(carla, m5), participatesIn(diego, m5), participatesIn(amanda, m5), participatesIn(thiago, m5),
participatesIn(renata, m5), participatesIn(guilherme, m5), participatesIn(lara, m5), participatesIn(anderson, m5),
participatesIn(manuel, m5), participatesIn(eduarda, m5), participatesIn(julio, m5), participatesIn(solange, m5),
participatesIn(lucia, m5), participatesIn(bruno, m5), participatesIn(bruna, m5), participatesIn(sonia, m5),
participatesIn(alfredo, m5), participatesIn(jorge, m5), participatesIn(yasmin, m5), participatesIn(alberto, m5),
participatesIn(luiza, m5), participatesIn(sofia, m5), participatesIn(gabriel, m5), participatesIn(manuel, m6),
participatesIn(eduarda, m6), participatesIn(leticia, m6), participatesIn(anderson, m6), participatesIn(solange, m6),
participatesIn(joaquim, m6), participatesIn(maria, m6), participatesIn(caio, m6), participatesIn(heloisa, m6),

participatesIn(marcelo, m6), participatesIn(carla, m6), participatesIn(diego, m6), participatesIn(amanda, m6),
participatesIn(thiago, m6), participatesIn(renata, m6), participatesIn(guilherme, m6), participatesIn(lara, m6),
participatesIn(roberto, m6), participatesIn(roberta, m6), participatesIn(augusto, m6), participatesIn(julieta, m6),
participatesIn(julio, m6), participatesIn(lucia, m6), participatesIn(bruno, m6), participatesIn(bruna, m6),
participatesIn(sonia, m6), participatesIn(alfredo, m6), participatesIn(jorge, m6), participatesIn(yasmin, m6),
participatesIn(alberto, m6), participatesIn(luiza, m6), participatesIn(sofia, m6), participatesIn(gabriel, m6),
participatesIn(julio, m7), participatesIn(lucia, m7), participatesIn(luiz, m7), participatesIn(augusto, m7),
participatesIn(julieta, m7), participatesIn(anderson, m7), participatesIn(solange, m7), participatesIn(caio, m7),
participatesIn(heloisa, m7), participatesIn(marcelo, m7), participatesIn(carla, m7), participatesIn(diego, m7),
participatesIn(amanda, m7), participatesIn(thiago, m7), participatesIn(renata, m7), participatesIn(guilherme, m7),
participatesIn(lara, m7), participatesIn(roberto, m7), participatesIn(roberta, m7), participatesIn(manuel, m7),
participatesIn(eduarda, m7), participatesIn(bruno, m7), participatesIn(joaquim, m7), participatesIn(maria, m7),
participatesIn(bruna, m7), participatesIn(sonia, m7), participatesIn(alfredo, m7), participatesIn(jorge, m7),
participatesIn(yasmin, m7), participatesIn(bruno, m8), participatesIn(bruna, m8), participatesIn(marcio, m8),
participatesIn(augusto, m8), participatesIn(julieta, m8), participatesIn(joaquim, m8), participatesIn(maria, m8),
participatesIn(anderson, m8), participatesIn(solange, m8), participatesIn(marcelo, m8), participatesIn(carla, m8),
participatesIn(diego, m8), participatesIn(amanda, m8), participatesIn(thiago, m8), participatesIn(renata, m8),
participatesIn(guilherme, m8), participatesIn(lara, m8), participatesIn(roberto, m8), participatesIn(roberta, m8),

participatesIn(manuel, m8), participatesIn(eduarda, m8), participatesIn(julio, m8), participatesIn(lucia, m8),
participatesIn(sonia, m8), participatesIn(alfredo, m8), participatesIn(caio, m8), participatesIn(heloisa, m8),
participatesIn(jorge, m8), participatesIn(yasmin, m8), participatesIn(alberto, m8), participatesIn(luiza, m8),
participatesIn(sofia, m8), participatesIn(gabriel, m8), participatesIn(marcelo, m9), participatesIn(sonia, m9),
participatesIn(leticia, m9), participatesIn(augusto, m9), participatesIn(julieta, m9), participatesIn(joaquim, m9),
participatesIn(maria, m9), participatesIn(caio, m9), participatesIn(heloisa, m9), participatesIn(roberto, m9),
participatesIn(carla, m9), participatesIn(diego, m9), participatesIn(amanda, m9), participatesIn(thiago, m9),
participatesIn(renata, m9), participatesIn(guilherme, m9), participatesIn(lara, m9), participatesIn(anderson, m9),
participatesIn(manuel, m9), participatesIn(eduarda, m9), participatesIn(julio, m9), participatesIn(solange, m9),
participatesIn(lucia, m9), participatesIn(bruno, m9), participatesIn(bruna, m9), participatesIn(roberta, m9),
participatesIn(alfredo, m9), participatesIn(jorge, m9), participatesIn(yasmin, m9), participatesIn(alberto, m9),
participatesIn(luiza, m9), participatesIn(sofia, m9), participatesIn(gabriel, m9), participatesIn(marcelo, m10),
participatesIn(alfredo, m10), participatesIn(luiz, m10), participatesIn(anderson, m10), participatesIn(joaquim, m10),
participatesIn(maria, m10), participatesIn(caio, m10), participatesIn(heloisa, m10), participatesIn(eduarda, m10),
participatesIn(carla, m10), participatesIn(diego, m10), participatesIn(amanda, m10), participatesIn(manuel, m10),
participatesIn(thiago, m10), participatesIn(renata, m10), participatesIn(guilherme, m10), participatesIn(lara, m10),
participatesIn(roberto, m10), participatesIn(roberta, m10), participatesIn(augusto, m10), participatesIn(julieta, m10),
participatesIn(julio, m10), participatesIn(lucia, m10), participatesIn(bruno, m10), participatesIn(bruna, m10),

participatesIn(sonia, m10), participatesIn(jorge, m10), participatesIn(yasmin, m10), participatesIn(solange, m10),
participatesIn(alberto, m10), participatesIn(luiza, m10), participatesIn(sofia, m10), participatesIn(gabriel, m10),
participatesIn(marcelo, m11), participatesIn(jorge, m11), participatesIn(marcio, m11), participatesIn(augusto, m11),
participatesIn(julieta, m11), participatesIn(joaquim, m11), participatesIn(maria, m11), participatesIn(caio, m11),
participatesIn(heloisa, m11), participatesIn(roberto, m11), participatesIn(sonia, m11), participatesIn(carla, m11),
participatesIn(diego, m11), participatesIn(amanda, m11), participatesIn(thiago, m11), participatesIn(renata, m11),
participatesIn(guilherme, m11), participatesIn(lara, m11), participatesIn(anderson, m11), participatesIn(manuel, m11),
participatesIn(eduarda, m11), participatesIn(julio, m11), participatesIn(solange, m11), participatesIn(lucia, m11),
participatesIn(bruno, m11), participatesIn(bruna, m11), participatesIn(roberta, m11), participatesIn(alfredo, m11),
participatesIn(yasmin, m11), participatesIn(alberto, m11), participatesIn(luiza, m11), participatesIn(sofia, m11),
participatesIn(gabriel, m11), participatesIn(marcelo, m12), participatesIn(yasmin, m12), participatesIn(leticia, m12),
participatesIn(anderson, m12), participatesIn(alfredo, m12), participatesIn(joaquim, m12), participatesIn(maria, m12),
participatesIn(caio, m12), participatesIn(heloisa, m12), participatesIn(eduarda, m12), participatesIn(carla, m12),
participatesIn(diego, m12), participatesIn(amanda, m12), participatesIn(manuel, m12), participatesIn(thiago, m12),
participatesIn(renata, m12), participatesIn(guilherme, m12), participatesIn(lara, m12), participatesIn(roberto, m12),
participatesIn(roberta, m12), participatesIn(augusto, m12), participatesIn(julieta, m12), participatesIn(julio, m12),
participatesIn(lucia, m12), participatesIn(bruno, m12), participatesIn(bruna, m12), participatesIn(sonia, m12),
participatesIn(jorge, m12), participatesIn(solange, m12), participatesIn(alberto, m12), participatesIn(luiza, m12),

participatesIn(sofia, m12), participatesIn(gabriel, m12), participatesIn(marcelo, m13), participatesIn(alberto, m13),
participatesIn(luiz, m13), participatesIn(anderson, m13), participatesIn(joaquim, m13), participatesIn(maria, m13),
participatesIn(caio, m13), participatesIn(heloisa, m13), participatesIn(eduarda, m13), participatesIn(carla, m13),
participatesIn(diego, m13), participatesIn(amanda, m13), participatesIn(manuel, m13), participatesIn(thiago, m13),
participatesIn(renata, m13), participatesIn(guilherme, m13), participatesIn(lara, m13), participatesIn(roberto, m13),
participatesIn(roberta, m13), participatesIn(augusto, m13), participatesIn(julieta, m13), participatesIn(julio, m13),
participatesIn(lucia, m13), participatesIn(bruno, m13), participatesIn(bruna, m13), participatesIn(sonia, m13),
participatesIn(jorge, m13), participatesIn(yasmin, m13), participatesIn(solange, m13), participatesIn(luiza, m13),
participatesIn(sofia, m13), participatesIn(alfredo, m13), participatesIn(gabriel, m13), participatesIn(marcelo, m14),
participatesIn(luiza, m14), participatesIn(marcio, m14), participatesIn(augusto, m14), participatesIn(julieta, m14),
participatesIn(joaquim, m14), participatesIn(maria, m14), participatesIn(caio, m14), participatesIn(heloisa, m14),
participatesIn(roberto, m14), participatesIn(sonia, m14), participatesIn(carla, m14), participatesIn(diego, m14),
participatesIn(amanda, m14), participatesIn(thiago, m14), participatesIn(renata, m14), participatesIn(guilherme, m14),
participatesIn(lara, m14), participatesIn(anderson, m14), participatesIn(manuel, m14), participatesIn(eduarda, m14),
participatesIn(julio, m14), participatesIn(solange, m14), participatesIn(lucia, m14), participatesIn(bruno, m14),
participatesIn(bruna, m14), participatesIn(roberta, m14), participatesIn(alfredo, m14), participatesIn(yasmin, m14),
participatesIn(alberto, m14), participatesIn(sofia, m14), participatesIn(jorge, m14), participatesIn(gabriel, m14),
participatesIn(marcelo, m15), participatesIn(gabriel, m15), participatesIn(leticia, m15), participatesIn(anderson, m15),

participatesIn(alfredo, m15), participatesIn(joaquim, m15), participatesIn(maria, m15), participatesIn(caio, m15),
participatesIn(heloisa, m15), participatesIn(eduarda, m15), participatesIn(carla, m15), participatesIn(diego, m15),
participatesIn(amanda, m15), participatesIn(manuel, m15), participatesIn(thiago, m15), participatesIn(renata, m15),
participatesIn(guilherme, m15), participatesIn(lara, m15), participatesIn(roberto, m15), participatesIn(roberta, m15),
participatesIn(augusto, m15), participatesIn(julieta, m15), participatesIn(julio, m15), participatesIn(lucia, m15),
participatesIn(bruno, m15), participatesIn(bruna, m15), participatesIn(sonia, m15), participatesIn(jorge, m15),
participatesIn(solange, m15), participatesIn(alberto, m15), participatesIn(luiza, m15), participatesIn(sofia, m15),
participatesIn(yasmin, m15), participatesIn(marcelo, m16), participatesIn(sofia, m16), participatesIn(luiz, m16),
participatesIn(anderson, m16), participatesIn(alfredo, m16), participatesIn(joaquim, m16), participatesIn(maria, m16),
participatesIn(caio, m16), participatesIn(heloisa, m16), participatesIn(eduarda, m16), participatesIn(carla, m16),
participatesIn(diego, m16), participatesIn(amanda, m16), participatesIn(manuel, m16), participatesIn(thiago, m16),
participatesIn(renata, m16), participatesIn(guilherme, m16), participatesIn(lara, m16), participatesIn(roberto, m16),
participatesIn(roberta, m16), participatesIn(augusto, m16), participatesIn(julieta, m16), participatesIn(julio, m16),
participatesIn(lucia, m16), participatesIn(bruno, m16), participatesIn(bruna, m16), participatesIn(sonia, m16),
participatesIn(jorge, m16), participatesIn(solange, m16), participatesIn(alberto, m16), participatesIn(luiza, m16),
participatesIn(gabriel, m16), participatesIn(yasmin, m16), participatesIn(carla, m17), participatesIn(sonia, m17),
participatesIn(marcio, m17), participatesIn(augusto, m17), participatesIn(julieta, m17), participatesIn(joaquim, m17),
participatesIn(maria, m17), participatesIn(caio, m17), participatesIn(heloisa, m17), participatesIn(roberto, m17),

participatesIn(marcelo, m17), participatesIn(diego, m17), participatesIn(amanda, m17), participatesIn(thiago, m17),
participatesIn(renata, m17), participatesIn(guilherme, m17), participatesIn(lara, m17), participatesIn(anderson, m17),
participatesIn(manuel, m17), participatesIn(eduarda, m17), participatesIn(julio, m17), participatesIn(solange, m17),
participatesIn(lucia, m17), participatesIn(bruno, m17), participatesIn(bruna, m17), participatesIn(roberta, m17),
participatesIn(alfredo, m17), participatesIn(jorge, m17), participatesIn(yasmin, m17), participatesIn(alberto, m17),
participatesIn(luiza, m17), participatesIn(sofia, m17), participatesIn(gabriel, m17), participatesIn(carla, m18),
participatesIn(alfredo, m18), participatesIn(leticia, m18), participatesIn(anderson, m18), participatesIn(joaquim, m18),
participatesIn(maria, m18), participatesIn(caio, m18), participatesIn(heloisa, m18), participatesIn(eduarda, m18),
participatesIn(marcelo, m18), participatesIn(diego, m18), participatesIn(amanda, m18), participatesIn(manuel, m18),
participatesIn(thiago, m18), participatesIn(renata, m18), participatesIn(guilherme, m18), participatesIn(lara, m18),
participatesIn(roberto, m18), participatesIn(roberta, m18), participatesIn(augusto, m18), participatesIn(julieta, m18),
participatesIn(julio, m18), participatesIn(lucia, m18), participatesIn(bruno, m18), participatesIn(bruna, m18),
participatesIn(sonia, m18), participatesIn(jorge, m18), participatesIn(yasmin, m18), participatesIn(solange, m18),
participatesIn(alberto, m18), participatesIn(luiza, m18), participatesIn(sofia, m18), participatesIn(gabriel, m18),
participatesIn(carla, m19), participatesIn(jorge, m19), participatesIn(luiz, m19), participatesIn(augusto, m19),
participatesIn(julieta, m19), participatesIn(joaquim, m19), participatesIn(maria, m19), participatesIn(caio, m19),
participatesIn(heloisa, m19), participatesIn(roberto, m19), participatesIn(sonia, m19), participatesIn(marcelo, m19),
participatesIn(diego, m19), participatesIn(amanda, m19), participatesIn(thiago, m19), participatesIn(renata, m19),

participatesIn(guilherme, m19), participatesIn(lara, m19), participatesIn(anderson, m19), participatesIn(manuel, m19),
participatesIn(eduarda, m19), participatesIn(julio, m19), participatesIn(solange, m19), participatesIn(lucia, m19),
participatesIn(bruno, m19), participatesIn(bruna, m19), participatesIn(roberta, m19), participatesIn(alfredo, m19),
participatesIn(yasmin, m19), participatesIn(alberto, m19), participatesIn(luiza, m19), participatesIn(sofia, m19),
participatesIn(gabriel, m19), participatesIn(carla, m20), participatesIn(yasmin, m20), participatesIn(marcio, m20),
participatesIn(anderson, m20), participatesIn(alfredo, m20), participatesIn(joaquim, m20), participatesIn(maria, m20),
participatesIn(caio, m20), participatesIn(heloisa, m20), participatesIn(eduarda, m20), participatesIn(marcelo, m20),
participatesIn(diego, m20), participatesIn(amanda, m20), participatesIn(manuel, m20), participatesIn(thiago, m20),
participatesIn(renata, m20), participatesIn(guilherme, m20), participatesIn(lara, m20), participatesIn(roberto, m20),
participatesIn(roberta, m20), participatesIn(augusto, m20), participatesIn(julieta, m20), participatesIn(julio, m20),
participatesIn(lucia, m20), participatesIn(bruno, m20), participatesIn(bruna, m20), participatesIn(sonia, m20),
participatesIn(jorge, m20), participatesIn(solange, m20), participatesIn(alberto, m20), participatesIn(luiza, m20),
participatesIn(sofia, m20), participatesIn(gabriel, m20), participatesIn(carla, m21), participatesIn(alberto, m21),
participatesIn(leticia, m21), participatesIn(anderson, m21), participatesIn(joaquim, m21), participatesIn(maria, m21),
participatesIn(caio, m21), participatesIn(heloisa, m21), participatesIn(eduarda, m21), participatesIn(marcelo, m21),
participatesIn(diego, m21), participatesIn(amanda, m21), participatesIn(manuel, m21), participatesIn(thiago, m21),
participatesIn(renata, m21), participatesIn(guilherme, m21), participatesIn(lara, m21), participatesIn(roberto, m21),
participatesIn(roberta, m21), participatesIn(augusto, m21), participatesIn(julieta, m21), participatesIn(julio, m21),

participatesIn(lucia, m21), participatesIn(bruno, m21), participatesIn(bruna, m21), participatesIn(sonia, m21),
participatesIn(jorge, m21), participatesIn(yasmin, m21), participatesIn(solange, m21), participatesIn(luiza, m21),
participatesIn(sofia, m21), participatesIn(alfredo, m21), participatesIn(gabriel, m21), participatesIn(carla, m22),
participatesIn(luiza, m22), participatesIn(luiz, m22), participatesIn(augusto, m22), participatesIn(julieta, m22),
participatesIn(joaquim, m22), participatesIn(maria, m22), participatesIn(caio, m22), participatesIn(heloisa, m22),
participatesIn(roberto, m22), participatesIn(sonia, m22), participatesIn(marcelo, m22), participatesIn(diego, m22),
participatesIn(amanda, m22), participatesIn(thiago, m22), participatesIn(renata, m22), participatesIn(guilherme, m22),
participatesIn(lara, m22), participatesIn(anderson, m22), participatesIn(manuel, m22), participatesIn(eduarda, m22),
participatesIn(julio, m22), participatesIn(solange, m22), participatesIn(lucia, m22), participatesIn(bruno, m22),
participatesIn(bruna, m22), participatesIn(roberta, m22), participatesIn(alfredo, m22), participatesIn(yasmin, m22),
participatesIn(alberto, m22), participatesIn(sofia, m22), participatesIn(jorge, m22), participatesIn(gabriel, m22),
participatesIn(carla, m23), participatesIn(gabriel, m23), participatesIn(marcio, m23), participatesIn(anderson, m23),
participatesIn(alfredo, m23), participatesIn(joaquim, m23), participatesIn(maria, m23), participatesIn(caio, m23),
participatesIn(heloisa, m23), participatesIn(eduarda, m23), participatesIn(marcelo, m23), participatesIn(diego, m23),
participatesIn(amanda, m23), participatesIn(manuel, m23), participatesIn(thiago, m23), participatesIn(renata, m23),
participatesIn(guilherme, m23), participatesIn(lara, m23), participatesIn(roberto, m23), participatesIn(roberta, m23),
participatesIn(augusto, m23), participatesIn(julieta, m23), participatesIn(julio, m23), participatesIn(lucia, m23),
participatesIn(bruno, m23), participatesIn(bruna, m23), participatesIn(sonia, m23), participatesIn(jorge, m23),

participatesIn(solange, m23), participatesIn(alberto, m23), participatesIn(luiza, m23), participatesIn(sofia, m23),
participatesIn(yasmin, m23), participatesIn(carla, m24), participatesIn(sofia, m24), participatesIn(leticia, m24),
participatesIn(anderson, m24), participatesIn(alfredo, m24), participatesIn(joaquim, m24), participatesIn(maria, m24),
participatesIn(caio, m24), participatesIn(heloisa, m24), participatesIn(eduarda, m24), participatesIn(marcelo, m24),
participatesIn(diego, m24), participatesIn(amanda, m24), participatesIn(manuel, m24), participatesIn(thiago, m24),
participatesIn(renata, m24), participatesIn(guilherme, m24), participatesIn(lara, m24), participatesIn(roberto, m24),
participatesIn(roberta, m24), participatesIn(augusto, m24), participatesIn(julieta, m24), participatesIn(julio, m24),
participatesIn(lucia, m24), participatesIn(bruno, m24), participatesIn(bruna, m24), participatesIn(sonia, m24),
participatesIn(jorge, m24), participatesIn(solange, m24), participatesIn(alberto, m24), participatesIn(luiza, m24),
participatesIn(gabriel, m24), participatesIn(yasmin, m24), participatesIn(diego, m25), participatesIn(sonia, m25),
participatesIn(luiz, m25), participatesIn(augusto, m25), participatesIn(julieta, m25), participatesIn(joaquim, m25),
participatesIn(maria, m25), participatesIn(caio, m25), participatesIn(heloisa, m25), participatesIn(roberto, m25),
participatesIn(marcelo, m25), participatesIn(carla, m25), participatesIn(amanda, m25), participatesIn(thiago, m25),
participatesIn(renata, m25), participatesIn(guilherme, m25), participatesIn(lara, m25), participatesIn(anderson, m25),
participatesIn(manuel, m25), participatesIn(eduarda, m25), participatesIn(julio, m25), participatesIn(solange, m25),
participatesIn(lucia, m25), participatesIn(bruno, m25), participatesIn(bruna, m25), participatesIn(roberta, m25),
participatesIn(alfredo, m25), participatesIn(jorge, m25), participatesIn(yasmin, m25), participatesIn(alberto, m25),
participatesIn(luiza, m25), participatesIn(sofia, m25), participatesIn(gabriel, m25), participatesIn(diego, m26),

participatesIn(alfredo, m26), participatesIn(marcio, m26), participatesIn(anderson, m26), participatesIn(joaquim, m26),

participatesIn(maria, m26), participatesIn(caio, m26), participatesIn(heloisa, m26), participatesIn(eduarda, m26),

participatesIn(marcelo, m26), participatesIn(carla, m26), participatesIn(amanda, m26), participatesIn(manuel, m26),

participatesIn(thiago, m26), participatesIn(renata, m26), participatesIn(guilherme, m26), participatesIn(lara, m26),

participatesIn(roberto, m26), participatesIn(roberta, m26), participatesIn(augusto, m26), participatesIn(julieta, m26),

participatesIn(julio, m26), participatesIn(lucia, m26), participatesIn(bruno, m26), participatesIn(bruna, m26),

participatesIn(sonia, m26), participatesIn(jorge, m26), participatesIn(yasmin, m26), participatesIn(solange, m26),

participatesIn(alberto, m26), participatesIn(luiza, m26), participatesIn(sofia, m26), participatesIn(gabriel, m26),

participatesIn(diego, m27), participatesIn(jorge, m27), participatesIn(leticia, m27), participatesIn(augusto, m27),

participatesIn(julieta, m27), participatesIn(joaquim, m27), participatesIn(maria, m27), participatesIn(caio, m27),

participatesIn(heloisa, m27), participatesIn(roberto, m27), participatesIn(sonia, m27), participatesIn(marcelo, m27),

participatesIn(carla, m27), participatesIn(amanda, m27), participatesIn(thiago, m27), participatesIn(renata, m27),

participatesIn(guilherme, m27), participatesIn(lara, m27), participatesIn(anderson, m27), participatesIn(manuel, m27),

participatesIn(eduarda, m27), participatesIn(julio, m27), participatesIn(solange, m27), participatesIn(lucia, m27),

participatesIn(bruno, m27), participatesIn(bruna, m27), participatesIn(roberta, m27), participatesIn(alfredo, m27),

participatesIn(yasmin, m27), participatesIn(alberto, m27), participatesIn(luiza, m27), participatesIn(sofia, m27),

participatesIn(gabriel, m27), participatesIn(diego, m28), participatesIn(yasmin, m28), participatesIn(luiz, m28),

participatesIn(anderson, m28), participatesIn(alfredo, m28), participatesIn(joaquim, m28), participatesIn(maria, m28),

participatesIn(caio, m28), participatesIn(heloisa, m28), participatesIn(eduarda, m28), participatesIn(marcelo, m28),
participatesIn(carla, m28), participatesIn(amanda, m28), participatesIn(manuel, m28), participatesIn(thiago, m28),
participatesIn(renata, m28), participatesIn(guilherme, m28), participatesIn(lara, m28), participatesIn(roberto, m28),
participatesIn(roberta, m28), participatesIn(augusto, m28), participatesIn(julieta, m28), participatesIn(julio, m28),
participatesIn(lucia, m28), participatesIn(bruno, m28), participatesIn(bruna, m28), participatesIn(sonia, m28),
participatesIn(jorge, m28), participatesIn(solange, m28), participatesIn(alberto, m28), participatesIn(luiza, m28),
participatesIn(sofia, m28), participatesIn(gabriel, m28), participatesIn(diego, m29), participatesIn(alberto, m29),
participatesIn(marcio, m29), participatesIn(anderson, m29), participatesIn(joaquim, m29), participatesIn(maria, m29),
participatesIn(caio, m29), participatesIn(heloisa, m29), participatesIn(eduarda, m29), participatesIn(marcelo, m29),
participatesIn(carla, m29), participatesIn(amanda, m29), participatesIn(manuel, m29), participatesIn(thiago, m29),
participatesIn(renata, m29), participatesIn(guilherme, m29), participatesIn(lara, m29), participatesIn(roberto, m29),
participatesIn(roberta, m29), participatesIn(augusto, m29), participatesIn(julieta, m29), participatesIn(julio, m29),
participatesIn(lucia, m29), participatesIn(bruno, m29), participatesIn(bruna, m29), participatesIn(sonia, m29),
participatesIn(jorge, m29), participatesIn(yasmin, m29), participatesIn(solange, m29), participatesIn(luiza, m29),
participatesIn(sofia, m29), participatesIn(alfredo, m29), participatesIn(gabriel, m29), participatesIn(diego, m30),
participatesIn(luiza, m30), participatesIn(leticia, m30), participatesIn(augusto, m30), participatesIn(julieta, m30),
participatesIn(joaquim, m30), participatesIn(maria, m30), participatesIn(caio, m30), participatesIn(heloisa, m30),
participatesIn(roberto, m30), participatesIn(sonia, m30), participatesIn(marcelo, m30), participatesIn(carla, m30),

participatesIn(amanda, m30), participatesIn(thiago, m30), participatesIn(renata, m30), participatesIn(guilherme, m30),
participatesIn(lara, m30), participatesIn(anderson, m30), participatesIn(manuel, m30), participatesIn(eduarda, m30),
participatesIn(julio, m30), participatesIn(solange, m30), participatesIn(lucia, m30), participatesIn(bruno, m30),
participatesIn(bruna, m30), participatesIn(roberta, m30), participatesIn(alfredo, m30), participatesIn(yasmin, m30),
participatesIn(alberto, m30), participatesIn(sofia, m30), participatesIn(jorge, m30), participatesIn(gabriel, m30),
participatesIn(diego, m31), participatesIn(gabriel, m31), participatesIn(luiz, m31), participatesIn(anderson, m31),
participatesIn(alfredo, m31), participatesIn(joaquim, m31), participatesIn(maria, m31), participatesIn(caio, m31),
participatesIn(heloisa, m31), participatesIn(eduarda, m31), participatesIn(marcelo, m31), participatesIn(carla, m31),
participatesIn(amanda, m31), participatesIn(manuel, m31), participatesIn(thiago, m31), participatesIn(renata, m31),
participatesIn(guilherme, m31), participatesIn(lara, m31), participatesIn(roberto, m31), participatesIn(roberta, m31),
participatesIn(augusto, m31), participatesIn(julieta, m31), participatesIn(julio, m31), participatesIn(lucia, m31),
participatesIn(bruno, m31), participatesIn(bruna, m31), participatesIn(sonia, m31), participatesIn(jorge, m31),
participatesIn(solange, m31), participatesIn(alberto, m31), participatesIn(luiza, m31), participatesIn(sofia, m31),
participatesIn(yasmin, m31), participatesIn(diego, m32), participatesIn(sofia, m32), participatesIn(marcio, m32),
participatesIn(anderson, m32), participatesIn(alfredo, m32), participatesIn(joaquim, m32), participatesIn(maria, m32),
participatesIn(caio, m32), participatesIn(heloisa, m32), participatesIn(eduarda, m32), participatesIn(marcelo, m32),
participatesIn(diego, m32), participatesIn(amanda, m32), participatesIn(manuel, m32), participatesIn(thiago, m32),
participatesIn(renata, m32), participatesIn(guilherme, m32), participatesIn(lara, m32), participatesIn(roberto, m32),

participatesIn(roberta, m32), participatesIn(augusto, m32), participatesIn(julieta, m32), participatesIn(julio, m32),
participatesIn(lucia, m32), participatesIn(bruno, m32), participatesIn(bruna, m32), participatesIn(sonia, m32),
participatesIn(jorge, m32), participatesIn(solange, m32), participatesIn(alberto, m32), participatesIn(luiza, m32),
participatesIn(gabriel, m32), participatesIn(yasmin, m32), participatesIn(amanda, m33), participatesIn(sonia, m33),
participatesIn(leticia, m33), participatesIn(augusto, m33), participatesIn(julieta, m33), participatesIn(joaquim, m33),
participatesIn(maria, m33), participatesIn(caio, m33), participatesIn(heloisa, m33), participatesIn(roberto, m33),
participatesIn(carla, m33), participatesIn(diego, m33), participatesIn(marcelo, m33), participatesIn(thiago, m33),
participatesIn(renata, m33), participatesIn(guilherme, m33), participatesIn(lara, m33), participatesIn(anderson, m33),
participatesIn(manuel, m33), participatesIn(eduarda, m33), participatesIn(julio, m33), participatesIn(solange, m33),
participatesIn(lucia, m33), participatesIn(bruno, m33), participatesIn(bruna, m33), participatesIn(roberta, m33),
participatesIn(alfredo, m33), participatesIn(jorge, m33), participatesIn(yasmin, m33), participatesIn(alberto, m33),
participatesIn(luiza, m33), participatesIn(sofia, m33), participatesIn(gabriel, m33), participatesIn(amanda, m34),
participatesIn(alfredo, m34), participatesIn(luiz, m34), participatesIn(anderson, m34), participatesIn(joaquim, m34),
participatesIn(maria, m34), participatesIn(caio, m34), participatesIn(heloisa, m34), participatesIn(eduarda, m34),
participatesIn(carla, m34), participatesIn(diego, m34), participatesIn(marcelo, m34), participatesIn(manuel, m34),
participatesIn(thiago, m34), participatesIn(renata, m34), participatesIn(guilherme, m34), participatesIn(lara, m34),
participatesIn(roberto, m34), participatesIn(roberta, m34), participatesIn(augusto, m34), participatesIn(julieta, m34),
participatesIn(julio, m34), participatesIn(lucia, m34), participatesIn(bruno, m34), participatesIn(bruna, m34),

participatesIn(sonia, m34), participatesIn(jorge, m34), participatesIn(yasmin, m34), participatesIn(solange, m34),
participatesIn(alberto, m34), participatesIn(luiza, m34), participatesIn(sofia, m34), participatesIn(gabriel, m34),
participatesIn(amanda, m35), participatesIn(jorge, m35), participatesIn(marcio, m35), participatesIn(augusto, m35),
participatesIn(julieta, m35), participatesIn(joaquim, m35), participatesIn(maria, m35), participatesIn(caio, m35),
participatesIn(heloisa, m35), participatesIn(roberto, m35), participatesIn(sonia, m35), participatesIn(carla, m35),
participatesIn(diego, m35), participatesIn(marcelo, m35), participatesIn(thiago, m35), participatesIn(renata, m35),
participatesIn(guilherme, m35), participatesIn(lara, m35), participatesIn(anderson, m35), participatesIn(manuel, m35),
participatesIn(eduarda, m35), participatesIn(julio, m35), participatesIn(solange, m35), participatesIn(lucia, m35),
participatesIn(bruno, m35), participatesIn(bruna, m35), participatesIn(roberta, m35), participatesIn(alfredo, m35),
participatesIn(yasmin, m35), participatesIn(alberto, m35), participatesIn(luiza, m35), participatesIn(sofia, m35),
participatesIn(gabriel, m35), participatesIn(amanda, m36), participatesIn(yasmin, m36), participatesIn(leticia, m36),
participatesIn(anderson, m36), participatesIn(alfredo, m36), participatesIn(joaquim, m36), participatesIn(maria, m36),
participatesIn(caio, m36), participatesIn(heloisa, m36), participatesIn(eduarda, m36), participatesIn(carla, m36),
participatesIn(diego, m36), participatesIn(marcelo, m36), participatesIn(manuel, m36), participatesIn(thiago, m36),
participatesIn(renata, m36), participatesIn(guilherme, m36), participatesIn(lara, m36), participatesIn(roberto, m36),
participatesIn(roberta, m36), participatesIn(augusto, m36), participatesIn(julieta, m36), participatesIn(julio, m36),
participatesIn(lucia, m36), participatesIn(bruno, m36), participatesIn(bruna, m36), participatesIn(sonia, m36),
participatesIn(jorge, m36), participatesIn(solange, m36), participatesIn(alberto, m36), participatesIn(luiza, m36),

participatesIn(sofia, m36), participatesIn(gabriel, m36), participatesIn(amanda, m37), participatesIn(alberto, m37),

participatesIn(luiz, m37), participatesIn(anderson, m37), participatesIn(joaquim, m37), participatesIn(maria, m37),

participatesIn(caio, m37), participatesIn(heloisa, m37), participatesIn(eduarda, m37), participatesIn(carla, m37),

participatesIn(diego, m37), participatesIn(marcelo, m37), participatesIn(manuel, m37), participatesIn(thiago, m37),

participatesIn(renata, m37), participatesIn(guilherme, m37), participatesIn(lara, m37), participatesIn(roberto, m37),

participatesIn(roberta, m37), participatesIn(augusto, m37), participatesIn(julieta, m37), participatesIn(julio, m37),

participatesIn(lucia, m37), participatesIn(bruno, m37), participatesIn(bruna, m37), participatesIn(sonia, m37),

participatesIn(jorge, m37), participatesIn(yasmin, m37), participatesIn(solange, m37), participatesIn(luiza, m37),

participatesIn(sofia, m37), participatesIn(alfredo, m37), participatesIn(gabriel, m37), participatesIn(amanda, m38),

participatesIn(luiza, m38), participatesIn(marcio, m38), participatesIn(augusto, m38), participatesIn(julieta, m38),

participatesIn(joaquim, m38), participatesIn(maria, m38), participatesIn(caio, m38), participatesIn(heloisa, m38),

participatesIn(roberto, m38), participatesIn(sonia, m38), participatesIn(carla, m38), participatesIn(diego, m38),

participatesIn(marcelo, m38), participatesIn(thiago, m38), participatesIn(renata, m38), participatesIn(guilherme, m38),

participatesIn(lara, m38), participatesIn(anderson, m38), participatesIn(manuel, m38), participatesIn(eduarda, m38),

participatesIn(julio, m38), participatesIn(solange, m38), participatesIn(lucia, m38), participatesIn(bruno, m38),

participatesIn(bruna, m38), participatesIn(roberta, m38), participatesIn(alfredo, m38), participatesIn(yasmin, m38),

participatesIn(alberto, m38), participatesIn(sofia, m38), participatesIn(jorge, m38), participatesIn(gabriel, m38),

participatesIn(amanda, m39), participatesIn(gabriel, m39), participatesIn(leticia, m39), participatesIn(anderson, m39),

participatesIn(alfredo, m39), participatesIn(joaquim, m39), participatesIn(maria, m39), participatesIn(caio, m39),
participatesIn(heloisa, m39), participatesIn(eduarda, m39), participatesIn(carla, m39), participatesIn(diego, m39),
participatesIn(marcelo, m39), participatesIn(manuel, m39), participatesIn(thiago, m39), participatesIn(renata, m39),
participatesIn(guilherme, m39), participatesIn(lara, m39), participatesIn(roberto, m39), participatesIn(roberta, m39),
participatesIn(augusto, m39), participatesIn(julieta, m39), participatesIn(julio, m39), participatesIn(lucia, m39),
participatesIn(bruno, m39), participatesIn(bruna, m39), participatesIn(sonia, m39), participatesIn(jorge, m39),
participatesIn(solange, m39), participatesIn(alberto, m39), participatesIn(luiza, m39), participatesIn(sofia, m39),
participatesIn(yasmin, m39), participatesIn(amanda, m40), participatesIn(sofia, m40), participatesIn(luiz, m40),
participatesIn(anderson, m40), participatesIn(alfredo, m40), participatesIn(joaquim, m40), participatesIn(maria, m40),
participatesIn(caio, m40), participatesIn(heloisa, m40), participatesIn(eduarda, m40), participatesIn(carla, m40),
participatesIn(diego, m40), participatesIn(marcelo, m40), participatesIn(manuel, m40), participatesIn(thiago, m40),
participatesIn(renata, m40), participatesIn(guilherme, m40), participatesIn(lara, m40), participatesIn(roberto, m40),
participatesIn(roberta, m40), participatesIn(augusto, m40), participatesIn(julieta, m40), participatesIn(julio, m40),
participatesIn(lucia, m40), participatesIn(bruno, m40), participatesIn(bruna, m40), participatesIn(sonia, m40),
participatesIn(jorge, m40), participatesIn(solange, m40), participatesIn(alberto, m40), participatesIn(luiza, m40)
participatesIn(gabriel, m40), participatesIn(yasmin, m40), participatesIn(thiago, m41), participatesIn(sonia, m41),
participatesIn(marcio, m41), participatesIn(augusto, m41), participatesIn(julieta, m41), participatesIn(joaquim, m41),
participatesIn(maria, m41), participatesIn(caio, m41), participatesIn(heloisa, m41), participatesIn(roberto, m41),

participatesIn(carla, m41), participatesIn(diego, m41), participatesIn(marcelo, m41), participatesIn(amanda, m41),
participatesIn(renata, m41), participatesIn(guilherme, m41), participatesIn(lara, m41), participatesIn(anderson, m41),
participatesIn(manuel, m41), participatesIn(eduarda, m41), participatesIn(julio, m41), participatesIn(solange, m41),
participatesIn(lucia, m41), participatesIn(bruno, m41), participatesIn(bruna, m41), participatesIn(roberta, m41),
participatesIn(alfredo, m41), participatesIn(jorge, m41), participatesIn(yasmin, m41), participatesIn(alberto, m41),
participatesIn(luiza, m41), participatesIn(sofia, m41), participatesIn(gabriel, m41), participatesIn(thiago, m42),
participatesIn(alfredo, m42), participatesIn(leticia, m42), participatesIn(anderson, m42), participatesIn(joaquim, m42),
participatesIn(maria, m42), participatesIn(caio, m42), participatesIn(heloisa, m42), participatesIn(eduarda, m42),
participatesIn(carla, m42), participatesIn(diego, m42), participatesIn(marcelo, m42), participatesIn(manuel, m42),
participatesIn(amanda, m42), participatesIn(renata, m42), participatesIn(guilherme, m42), participatesIn(lara, m42),
participatesIn(roberto, m42), participatesIn(roberta, m42), participatesIn(augusto, m42), participatesIn(julieta, m42),
participatesIn(julio, m42), participatesIn(lucia, m42), participatesIn(bruno, m42), participatesIn(bruna, m42),
participatesIn(sonia, m42), participatesIn(jorge, m42), participatesIn(yasmin, m42), participatesIn(solange, m42),
participatesIn(alberto, m42), participatesIn(luiza, m42), participatesIn(sofia, m42), participatesIn(gabriel, m42),
participatesIn(thiago, m43), participatesIn(jorge, m43), participatesIn(luiz, m43), participatesIn(augusto, m43),
participatesIn(julieta, m43), participatesIn(joaquim, m43), participatesIn(maria, m43), participatesIn(caio, m43),
participatesIn(heloisa, m43), participatesIn(roberto, m43), participatesIn(sonia, m43), participatesIn(carla, m43),
participatesIn(diego, m43), participatesIn(marcelo, m43), participatesIn(amanda, m43), participatesIn(renata, m43),

participatesIn(guilherme, m43), participatesIn(lara, m43), participatesIn(anderson, m43), participatesIn(manuel, m43),
participatesIn(eduarda, m43), participatesIn(julio, m43), participatesIn(solange, m43), participatesIn(lucia, m43),
participatesIn(bruno, m43), participatesIn(bruna, m43), participatesIn(roberta, m43), participatesIn(alfredo, m43),
participatesIn(yasmin, m43), participatesIn(alberto, m43), participatesIn(luiza, m43), participatesIn(sofia, m43),
participatesIn(gabriel, m43), participatesIn(thiago, m44), participatesIn(yasmin, m44), participatesIn(marcio, m44),
participatesIn(anderson, m44), participatesIn(alfredo, m44), participatesIn(joaquim, m44), participatesIn(maria, m44),
participatesIn(caio, m44), participatesIn(heloisa, m44), participatesIn(eduarda, m44), participatesIn(carla, m44),
participatesIn(diego, m44), participatesIn(marcelo, m44), participatesIn(manuel, m44), participatesIn(amanda, m44),
participatesIn(renata, m44), participatesIn(guilherme, m44), participatesIn(lara, m44), participatesIn(roberto, m44),
participatesIn(roberta, m44), participatesIn(augusto, m44), participatesIn(julieta, m44), participatesIn(julio, m44),
participatesIn(lucia, m44), participatesIn(bruno, m44), participatesIn(bruna, m44), participatesIn(sonia, m44),
participatesIn(jorge, m44), participatesIn(solange, m44), participatesIn(alberto, m44), participatesIn(luiza, m44),
participatesIn(sofia, m44), participatesIn(gabriel, m44), participatesIn(thiago, m45), participatesIn(alberto, m45),
participatesIn(leticia, m45), participatesIn(anderson, m45), participatesIn(joaquim, m45), participatesIn(maria, m45),
participatesIn(caio, m45), participatesIn(heloisa, m45), participatesIn(eduarda, m45), participatesIn(carla, m45),
participatesIn(diego, m45), participatesIn(marcelo, m45), participatesIn(manuel, m45), participatesIn(amanda, m45),
participatesIn(renata, m45), participatesIn(guilherme, m45), participatesIn(lara, m45), participatesIn(roberto, m45),
participatesIn(roberta, m45), participatesIn(augusto, m45), participatesIn(julieta, m45), participatesIn(julio, m45),

participatesIn(lucia, m45), participatesIn(bruno, m45), participatesIn(bruna, m45), participatesIn(sonia, m45),
participatesIn(jorge, m45), participatesIn(yasmin, m45), participatesIn(solange, m45), participatesIn(luiza, m45),
participatesIn(sofia, m45), participatesIn(alfredo, m45), participatesIn(gabriel, m45), participatesIn(thiago, m46),
participatesIn(luiza, m46), participatesIn(luiz, m46), participatesIn(augusto, m46), participatesIn(julieta, m46),
participatesIn(joaquim, m46), participatesIn(maria, m46), participatesIn(caio, m46), participatesIn(heloisa, m46),
participatesIn(roberto, m46), participatesIn(sonia, m46), participatesIn(carla, m46), participatesIn(diego, m46),
participatesIn(marcelo, m46), participatesIn(amanda, m46), participatesIn(renata, m46), participatesIn(guilherme, m46),
participatesIn(lara, m46), participatesIn(anderson, m46), participatesIn(manuel, m46), participatesIn(eduarda, m46),
participatesIn(julio, m46), participatesIn(solange, m46), participatesIn(lucia, m46), participatesIn(bruno, m46),
participatesIn(bruna, m46), participatesIn(roberta, m46), participatesIn(alfredo, m46), participatesIn(yasmin, m46),
participatesIn(alberto, m46), participatesIn(sofia, m46), participatesIn(jorge, m46), participatesIn(gabriel, m46),
participatesIn(thiago, m47), participatesIn(gabriel, m47), participatesIn(marcio, m47), participatesIn(anderson, m47),
participatesIn(alfredo, m47), participatesIn(joaquim, m47), participatesIn(maria, m47), participatesIn(caio, m47),
participatesIn(heloisa, m47), participatesIn(eduarda, m47), participatesIn(carla, m47), participatesIn(diego, m47),
participatesIn(marcelo, m47), participatesIn(manuel, m47), participatesIn(amanda, m47), participatesIn(renata, m47),
participatesIn(guilherme, m47), participatesIn(lara, m47), participatesIn(roberto, m47), participatesIn(roberta, m47),
participatesIn(augusto, m47), participatesIn(julieta, m47), participatesIn(julio, m47), participatesIn(lucia, m47),
participatesIn(bruno, m47), participatesIn(bruna, m47), participatesIn(sonia, m47), participatesIn(jorge, m47),

participatesIn(solange, m47), participatesIn(alberto, m47), participatesIn(luiza, m47), participatesIn(sofia, m47),
participatesIn(yasmin, m47), participatesIn(thiago, m48), participatesIn(sofia, m48), participatesIn(leticia, m48),
participatesIn(anderson, m48), participatesIn(alfredo, m48), participatesIn(joaquim, m48), participatesIn(maria, m48),
participatesIn(caio, m48), participatesIn(heloisa, m48), participatesIn(eduarda, m48), participatesIn(carla, m48),
participatesIn(diego, m48), participatesIn(marcelo, m48), participatesIn(manuel, m48), participatesIn(amanda, m48),
participatesIn(renata, m48), participatesIn(guilherme, m48), participatesIn(lara, m48), participatesIn(roberto, m48),
participatesIn(roberta, m48), participatesIn(augusto, m48), participatesIn(julieta, m48), participatesIn(julio, m48),
participatesIn(lucia, m48), participatesIn(bruno, m48), participatesIn(bruna, m48), participatesIn(sonia, m48),
participatesIn(jorge, m48), participatesIn(solange, m48), participatesIn(alberto, m48), participatesIn(luiza, m48),
participatesIn(gabriel, m48), participatesIn(yasmin, m48), participatesIn(renata, m49), participatesIn(sonia, m49),
participatesIn(luiz, m49), participatesIn(augusto, m49), participatesIn(julieta, m49), participatesIn(joaquim, m49),
participatesIn(maria, m49), participatesIn(caio, m49), participatesIn(heloisa, m49), participatesIn(roberto, m49),
participatesIn(carla, m49), participatesIn(diego, m49), participatesIn(marcelo, m49), participatesIn(amanda, m49),
participatesIn(thiago, m49), participatesIn(guilherme, m49), participatesIn(lara, m49), participatesIn(anderson, m49),
participatesIn(manuel, m49), participatesIn(eduarda, m49), participatesIn(julio, m49), participatesIn(solange, m49),
participatesIn(lucia, m49), participatesIn(bruno, m49), participatesIn(bruna, m49), participatesIn(roberta, m49),
participatesIn(alfredo, m49), participatesIn(jorge, m49), participatesIn(yasmin, m49), participatesIn(alberto, m49),
participatesIn(luiza, m49), participatesIn(sofia, m49), participatesIn(gabriel, m49), participatesIn(renata, m50),

participatesIn(alfredo, m50), participatesIn(marcio, m50), participatesIn(anderson, m50), participatesIn(joaquim, m50),
participatesIn(maria, m50), participatesIn(caio, m50), participatesIn(heloisa, m50), participatesIn(eduarda, m50),
participatesIn(carla, m50), participatesIn(diego, m50), participatesIn(marcelo, m50), participatesIn(manuel, m50),
participatesIn(amanda, m50), participatesIn(thiago, m50), participatesIn(guilherme, m50), participatesIn(lara, m50),
participatesIn(roberto, m50), participatesIn(roberta, m50), participatesIn(augusto, m50), participatesIn(julieta, m50),
participatesIn(julio, m50), participatesIn(lucia, m50), participatesIn(bruno, m50), participatesIn(bruna, m50),
participatesIn(sonia, m50), participatesIn(jorge, m50), participatesIn(yasmin, m50), participatesIn(solange, m50),
participatesIn(alberto, m50), participatesIn(luiza, m50), participatesIn(sofia, m50), participatesIn(gabriel, m50)
participatesIn(renata, m51), participatesIn(jorge, m51), participatesIn(leticia, m51), participatesIn(augusto, m51),
participatesIn(julieta, m51), participatesIn(joaquim, m51), participatesIn(maria, m51), participatesIn(caio, m51),
participatesIn(heloisa, m51), participatesIn(roberto, m51), participatesIn(sonia, m51), participatesIn(carla, m51),
participatesIn(diego, m51), participatesIn(marcelo, m51), participatesIn(amanda, m51), participatesIn(thiago, m51),
participatesIn(guilherme, m51), participatesIn(lara, m51), participatesIn(anderson, m51), participatesIn(manuel, m51),
participatesIn(eduarda, m51), participatesIn(julio, m51), participatesIn(solange, m51), participatesIn(lucia, m51),
participatesIn(bruno, m51), participatesIn(bruna, m51), participatesIn(roberta, m51), participatesIn(alfredo, m51),
participatesIn(yasmin, m51), participatesIn(alberto, m51), participatesIn(luiza, m51), participatesIn(sofia, m51),
participatesIn(gabriel, m51), participatesIn(renata, m52), participatesIn(yasmin, m52), participatesIn(luiz, m52),
participatesIn(anderson, m52), participatesIn(alfredo, m52), participatesIn(joaquim, m52), participatesIn(maria, m52),

participatesIn(caio, m52), participatesIn(heloisa, m52), participatesIn(eduarda, m52), participatesIn(carla, m52),
participatesIn(diego, m52), participatesIn(marcelo, m52), participatesIn(manuel, m52), participatesIn(amanda, m52),
participatesIn(thiago, m52), participatesIn(guilherme, m52), participatesIn(lara, m52), participatesIn(roberto, m52),
participatesIn(roberta, m52), participatesIn(augusto, m52), participatesIn(julieta, m52), participatesIn(julio, m52),
participatesIn(lucia, m52), participatesIn(bruno, m52), participatesIn(bruna, m52), participatesIn(sonia, m52),
participatesIn(jorge, m52), participatesIn(solange, m52), participatesIn(alberto, m52), participatesIn(luiza, m52),
participatesIn(sofia, m52), participatesIn(gabriel, m52), participatesIn(renata, m53), participatesIn(alberto, m53),
participatesIn(marcio, m53), participatesIn(anderson, m53), participatesIn(joaquim, m53), participatesIn(maria, m53),
participatesIn(caio, m53), participatesIn(heloisa, m53), participatesIn(eduarda, m53), participatesIn(carla, m53),
participatesIn(diego, m53), participatesIn(marcelo, m53), participatesIn(manuel, m53), participatesIn(amanda, m53),
participatesIn(thiago, m53), participatesIn(guilherme, m53), participatesIn(lara, m53), participatesIn(roberto, m53),
participatesIn(roberta, m53), participatesIn(augusto, m53), participatesIn(julieta, m53), participatesIn(julio, m53),
participatesIn(lucia, m53), participatesIn(bruno, m53), participatesIn(bruna, m53), participatesIn(sonia, m53),
participatesIn(jorge, m53), participatesIn(yasmin, m53), participatesIn(solange, m53), participatesIn(luiza, m53),
participatesIn(sofia, m53), participatesIn(alfredo, m53), participatesIn(gabriel, m53), participatesIn(renata, m54),
participatesIn(luiza, m54), participatesIn(leticia, m54), participatesIn(augusto, m54), participatesIn(julieta, m54),
participatesIn(joaquim, m54), participatesIn(maria, m54), participatesIn(caio, m54), participatesIn(heloisa, m54),
participatesIn(roberto, m54), participatesIn(sonia, m54), participatesIn(carla, m54), participatesIn(diego, m54),

participatesIn(marcelo, m54), participatesIn(amanda, m54), participatesIn(thiago, m54), participatesIn(guilherme, m54),

participatesIn(lara, m54), participatesIn(anderson, m54), participatesIn(manuel, m54), participatesIn(eduarda, m54),

participatesIn(julio, m54), participatesIn(solange, m54), participatesIn(lucia, m54), participatesIn(bruno, m54),

participatesIn(bruna, m54), participatesIn(roberta, m54), participatesIn(alfredo, m54), participatesIn(yasmin, m54),

participatesIn(alberto, m54), participatesIn(sofia, m54), participatesIn(jorge, m54), participatesIn(gabriel, m54),

participatesIn(renata, m55), participatesIn(gabriel, m55), participatesIn(luiz, m55), participatesIn(anderson, m55),

participatesIn(alfredo, m55), participatesIn(joaquim, m55), participatesIn(maria, m55), participatesIn(caio, m55),

participatesIn(heloisa, m55), participatesIn(eduarda, m55), participatesIn(carla, m55), participatesIn(diego, m55),

participatesIn(marcelo, m55), participatesIn(manuel, m55), participatesIn(amanda, m55), participatesIn(renata, m55),

participatesIn(guilherme, m55), participatesIn(lara, m55), participatesIn(roberto, m55), participatesIn(roberta, m55),

participatesIn(augusto, m55), participatesIn(julieta, m55), participatesIn(julio, m55), participatesIn(lucia, m55),

participatesIn(bruno, m55), participatesIn(bruna, m55), participatesIn(sonia, m55), participatesIn(jorge, m55),

participatesIn(solange, m55), participatesIn(alberto, m55), participatesIn(luiza, m55), participatesIn(sofia, m55),

participatesIn(yasmin, m55), participatesIn(renata, m56), participatesIn(sofia, m56), participatesIn(marcio, m56),

participatesIn(anderson, m56), participatesIn(alfredo, m56), participatesIn(joaquim, m56), participatesIn(maria, m56),

participatesIn(caio, m56), participatesIn(heloisa, m56), participatesIn(eduarda, m56), participatesIn(carla, m56),

participatesIn(diego, m56), participatesIn(marcelo, m56), participatesIn(manuel, m56), participatesIn(amanda, m56),

participatesIn(thiago, m56), participatesIn(guilherme, m56), participatesIn(lara, m56), participatesIn(roberto, m56),

participatesIn(roberta, m56), participatesIn(augusto, m56), participatesIn(julieta, m56), participatesIn(julio, m56),
participatesIn(lucia, m56), participatesIn(bruno, m56), participatesIn(bruna, m56), participatesIn(sonia, m56),
participatesIn(jorge, m56), participatesIn(solange, m56), participatesIn(alberto, m56), participatesIn(luiza, m56),
participatesIn(gabriel, m56), participatesIn(yasmin, m56), participatesIn(guilherme, m57), participatesIn(sonia, m57),
participatesIn(leticia, m57), participatesIn(augusto, m57), participatesIn(julieta, m57), participatesIn(joaquim, m57),
participatesIn(maria, m57), participatesIn(caio, m57), participatesIn(heloisa, m57), participatesIn(roberto, m57),
participatesIn(carla, m57), participatesIn(diego, m57), participatesIn(marcelo, m57), participatesIn(amanda, m57),
participatesIn(thiago, m57), participatesIn(renata, m57), participatesIn(lara, m57), participatesIn(anderson, m57),
participatesIn(manuel, m57), participatesIn(eduarda, m57), participatesIn(julio, m57), participatesIn(solange, m57),
participatesIn(lucia, m57), participatesIn(bruno, m57), participatesIn(bruna, m57), participatesIn(roberta, m57),
participatesIn(alfredo, m57), participatesIn(jorge, m57), participatesIn(yasmin, m57), participatesIn(alberto, m57),
participatesIn(luiza, m57), participatesIn(sofia, m57), participatesIn(gabriel, m57), participatesIn(guilherme, m58),
participatesIn(alfredo, m58), participatesIn(luiz, m58), participatesIn(anderson, m58), participatesIn(joaquim, m58),
participatesIn(maria, m58), participatesIn(caio, m58), participatesIn(heloisa, m58), participatesIn(eduarda, m58),
participatesIn(carla, m58), participatesIn(diego, m58), participatesIn(marcelo, m58), participatesIn(manuel, m58),
participatesIn(amanda, m58), participatesIn(thiago, m58), participatesIn(renata, m58), participatesIn(lara, m58),
participatesIn(roberto, m58), participatesIn(roberta, m58), participatesIn(augusto, m58), participatesIn(julieta, m58),
participatesIn(julio, m58), participatesIn(lucia, m58), participatesIn(bruno, m58), participatesIn(bruna, m58),

participatesIn(sonia, m58), participatesIn(jorge, m58), participatesIn(yasmin, m58), participatesIn(solange, m58),
participatesIn(alberto, m58), participatesIn(luiza, m58), participatesIn(sofia, m58), participatesIn(gabriel, m58),
participatesIn(guilherme, m59), participatesIn(jorge, m59), participatesIn(marcio, m59), participatesIn(augusto, m59),
participatesIn(julieta, m59), participatesIn(joaquim, m59), participatesIn(maria, m59), participatesIn(caio, m59),
participatesIn(heloisa, m59), participatesIn(roberto, m59), participatesIn(sonia, m59), participatesIn(carla, m59),
participatesIn(diego, m59), participatesIn(marcelo, m59), participatesIn(amanda, m59), participatesIn(thiago, m59),
participatesIn(renata, m59), participatesIn(lara, m59), participatesIn(anderson, m59), participatesIn(manuel, m59),
participatesIn(eduarda, m59), participatesIn(julio, m59), participatesIn(solange, m59), participatesIn(lucia, m59),
participatesIn(bruno, m59), participatesIn(bruna, m59), participatesIn(roberta, m59), participatesIn(alfredo, m59),
participatesIn(yasmin, m59), participatesIn(alberto, m59), participatesIn(luiza, m59), participatesIn(sofia, m59),
participatesIn(gabriel, m59), participatesIn(guilherme, m60), participatesIn(yasmin, m60), participatesIn(leticia, m60),
participatesIn(anderson, m60), participatesIn(alfredo, m60), participatesIn(joaquim, m60), participatesIn(maria, m60),
participatesIn(caio, m60), participatesIn(heloisa, m60), participatesIn(eduarda, m60), participatesIn(carla, m60),
participatesIn(diego, m60), participatesIn(marcelo, m60), participatesIn(manuel, m60), participatesIn(amanda, m60),
participatesIn(thiago, m60), participatesIn(renata, m60), participatesIn(lara, m60), participatesIn(roberto, m60),
participatesIn(roberta, m60), participatesIn(augusto, m60), participatesIn(julieta, m60), participatesIn(julio, m60),
participatesIn(lucia, m60), participatesIn(bruno, m60), participatesIn(bruna, m60), participatesIn(sonia, m60),
participatesIn(jorge, m60), participatesIn(solange, m60), participatesIn(alberto, m60), participatesIn(luiza, m60),

participatesIn(sofia, m60), participatesIn(gabriel, m60), participatesIn(guilherme, m61), participatesIn(alberto, m61),
participatesIn(luiz, m61), participatesIn(anderson, m61), participatesIn(joaquim, m61), participatesIn(maria, m61),
participatesIn(caio, m61), participatesIn(heloisa, m61), participatesIn(eduarda, m61), participatesIn(carla, m61),
participatesIn(diego, m61), participatesIn(marcelo, m61), participatesIn(manuel, m61), participatesIn(amanda, m61),
participatesIn(thiago, m61), participatesIn(renata, m61), participatesIn(lara, m61), participatesIn(roberto, m61),
participatesIn(roberta, m61), participatesIn(augusto, m61), participatesIn(julieta, m61), participatesIn(julio, m61),
participatesIn(lucia, m61), participatesIn(bruno, m61), participatesIn(bruna, m61), participatesIn(sonia, m61),
participatesIn(jorge, m61), participatesIn(yasmin, m61), participatesIn(solange, m61), participatesIn(luiza, m61),
participatesIn(sofia, m61), participatesIn(alfredo, m61), participatesIn(gabriel, m61), participatesIn(guilherme, m62),
participatesIn(luiza, m62), participatesIn(marcio, m62), participatesIn(augusto, m62), participatesIn(julieta, m62),
participatesIn(joaquim, m62), participatesIn(maria, m62), participatesIn(caio, m62), participatesIn(heloisa, m62),
participatesIn(roberto, m62), participatesIn(sonia, m62), participatesIn(carla, m62), participatesIn(diego, m62),
participatesIn(marcelo, m62), participatesIn(amanda, m62), participatesIn(thiago, m62), participatesIn(renata, m62),
participatesIn(lara, m62), participatesIn(anderson, m62), participatesIn(manuel, m62), participatesIn(eduarda, m62),
participatesIn(julio, m62), participatesIn(solange, m62), participatesIn(lucia, m62), participatesIn(bruno, m62),
participatesIn(bruna, m62), participatesIn(roberta, m62), participatesIn(alfredo, m62), participatesIn(yasmin, m62),
participatesIn(alberto, m62), participatesIn(sofia, m62), participatesIn(jorge, m62), participatesIn(gabriel, m62)
participatesIn(guilherme, m63), participatesIn(gabriel, m63), participatesIn(leticia, m63),
participatesIn(anderson, m63), participatesIn(alfredo, m63), participatesIn(joaquim, m63),
participatesIn(maria, m63),

participatesIn(caio, m63),
participatesIn(heloisa, m63), participatesIn(eduarda, m63), participatesIn(carla, m63), participatesIn(diego, m63),
participatesIn(marcelo, m63), participatesIn(manuel, m63), participatesIn(amanda, m63), participatesIn(renata, m63),
participatesIn(renata, m63), participatesIn(lara, m63), participatesIn(roberto, m63), participatesIn(roberta, m63),
participatesIn(augusto, m63), participatesIn(julieta, m63), participatesIn(julio, m63), participatesIn(lucia, m63),
participatesIn(bruno, m63), participatesIn(bruna, m63), participatesIn(sonia, m63), participatesIn(jorge, m63),
participatesIn(solange, m63), participatesIn(alberto, m63), participatesIn(luiza, m63), participatesIn(sofia, m63),
participatesIn(yasmin, m63), participatesIn(guilherme, m64), participatesIn(sofia, m64), participatesIn(luiz, m64),
participatesIn(anderson, m64), participatesIn(alfredo, m64), participatesIn(joaquim, m64), participatesIn(maria, m64),
participatesIn(caio, m64), participatesIn(heloisa, m64), participatesIn(eduarda, m64), participatesIn(carla, m64),
participatesIn(diego, m64), participatesIn(marcelo, m64), participatesIn(manuel, m64), participatesIn(amanda, m64),
participatesIn(thiago, m64), participatesIn(renata, m64), participatesIn(lara, m64), participatesIn(roberto, m64),
participatesIn(roberta, m64), participatesIn(augusto, m64), participatesIn(julieta, m64), participatesIn(julio, m64),
participatesIn(lucia, m64), participatesIn(bruno, m64), participatesIn(bruna, m64), participatesIn(sonia, m64),
participatesIn(jorge, m64), participatesIn(solange, m64), participatesIn(alberto, m64), participatesIn(luiza, m64),
participatesIn(gabriel, m64), participatesIn(yasmin, m64), participatesIn(lara, m65), participatesIn(sonia, m65),
participatesIn(marcio, m65), participatesIn(augusto, m65), participatesIn(julieta, m65), participatesIn(joaquim, m65),
participatesIn(maria, m65), participatesIn(caio, m65), participatesIn(heloisa, m65), participatesIn(roberto, m65),
participatesIn(carla, m65), participatesIn(diego, m65), participatesIn(marcelo, m65), participatesIn(amanda, m65),

participatesIn(thiago, m65), participatesIn(renata, m65), participatesIn(guilherme, m65), participatesIn(anderson, m65),

participatesIn(manuel, m65), participatesIn(eduarda, m65), participatesIn(julio, m65), participatesIn(solange, m65),

participatesIn(lucia, m65), participatesIn(bruno, m65), participatesIn(bruna, m65), participatesIn(roberta, m65),

participatesIn(alfredo, m65), participatesIn(jorge, m65), participatesIn(yasmin, m65), participatesIn(alberto, m65),

participatesIn(luiza, m65), participatesIn(sofia, m65), participatesIn(gabriel, m65), participatesIn(lara, m66),

participatesIn(alfredo, m66), participatesIn(leticia, m66), participatesIn(anderson, m66), participatesIn(joaquim, m66),

participatesIn(maria, m66), participatesIn(caio, m66), participatesIn(heloisa, m66), participatesIn(eduarda, m66),

participatesIn(carla, m66), participatesIn(diego, m66), participatesIn(marcelo, m66), participatesIn(manuel, m66),

participatesIn(amanda, m66), participatesIn(thiago, m66), participatesIn(renata, m66), participatesIn(guilherme, m66),

participatesIn(roberto, m66), participatesIn(roberta, m66), participatesIn(augusto, m66), participatesIn(julieta, m66),

participatesIn(julio, m66), participatesIn(lucia, m66), participatesIn(bruno, m66), participatesIn(bruna, m66),

participatesIn(sonia, m66), participatesIn(jorge, m66), participatesIn(yasmin, m66), participatesIn(solange, m66),

participatesIn(alberto, m66), participatesIn(luiza, m66), participatesIn(sofia, m66), participatesIn(gabriel, m66),

participatesIn(lara, m67), participatesIn(jorge, m67), participatesIn(luiz, m67), participatesIn(augusto, m67),

participatesIn(julieta, m67), participatesIn(joaquim, m67), participatesIn(maria, m67), participatesIn(caio, m67),

participatesIn(heloisa, m67), participatesIn(roberto, m67), participatesIn(sonia, m67), participatesIn(carla, m67),

participatesIn(diego, m67), participatesIn(marcelo, m67), participatesIn(amanda, m67), participatesIn(thiago, m67),

participatesIn(renata, m67), participatesIn(guilherme, m67), participatesIn(anderson, m67), participatesIn(manuel, m67), participatesIn(eduarda, m67), participatesIn(julio, m67), participatesIn(solange, m67), participatesIn(lucia, m67),

participatesIn(bruno, m67), participatesIn(bruna, m67), participatesIn(roberta, m67), participatesIn(alfredo, m67),

participatesIn(yasmin, m67), participatesIn(alberto, m67), participatesIn(luiza, m67), participatesIn(sofia, m67),

participatesIn(gabriel, m67), participatesIn(lara, m68), participatesIn(yasmin, m68), participatesIn(marcio, m68),

participatesIn(anderson, m68), participatesIn(alfredo, m68), participatesIn(joaquim, m68), participatesIn(maria, m68),

participatesIn(caio, m68), participatesIn(heloisa, m68), participatesIn(eduarda, m68), participatesIn(carla, m68),

participatesIn(diego, m68), participatesIn(marcelo, m68), participatesIn(manuel, m68), participatesIn(amanda, m68),

participatesIn(thiago, m68), participatesIn(renata, m68), participatesIn(guilherme, m68), participatesIn(roberto, m68),

participatesIn(roberta, m68), participatesIn(augusto, m68), participatesIn(julieta, m68), participatesIn(julio, m68),

participatesIn(lucia, m68), participatesIn(bruno, m68), participatesIn(bruna, m68), participatesIn(sonia, m68),

participatesIn(jorge, m68), participatesIn(solange, m68), participatesIn(alberto, m68), participatesIn(luiza, m68),

participatesIn(sofia, m68), participatesIn(gabriel, m68), participatesIn(lara, m69), participatesIn(alberto, m69),

participatesIn(leticia, m69), participatesIn(anderson, m69), participatesIn(joaquim, m69), participatesIn(maria, m69),

participatesIn(caio, m69), participatesIn(heloisa, m69), participatesIn(eduarda, m69), participatesIn(carla, m69),

participatesIn(diego, m69), participatesIn(marcelo, m69), participatesIn(manuel, m69), participatesIn(amanda, m69),

participatesIn(thiago, m69), participatesIn(renata, m69), participatesIn(guilherme, m69), participatesIn(roberto, m69),

participatesIn(roberta, m69), participatesIn(augusto, m69), participatesIn(julieta, m69), participatesIn(julio, m69),

participatesIn(lucia, m69), participatesIn(bruno, m69), participatesIn(bruna, m69), participatesIn(sonia, m69),

participatesIn(jorge, m69), participatesIn(yasmin, m69), participatesIn(solange, m69), participatesIn(luiza, m69),

participatesIn(sofia, m69), participatesIn(alfredo, m69), participatesIn(gabriel, m69), participatesIn(lara, m70),

participatesIn(luiza, m70), participatesIn(luiz, m70), participatesIn(augusto, m70), participatesIn(julieta, m70),

participatesIn(joaquim, m70), participatesIn(maria, m70), participatesIn(caio, m70), participatesIn(heloisa, m70),

participatesIn(roberto, m70), participatesIn(sonia, m70), participatesIn(carla, m70), participatesIn(diego, m70),

participatesIn(marcelo, m70), participatesIn(amanda, m70), participatesIn(thiago, m70), participatesIn(renata, m70),

participatesIn(guilherme, m70), participatesIn(anderson, m70), participatesIn(manuel, m70), participatesIn(eduarda, m70),

participatesIn(julio, m70), participatesIn(solange, m70), participatesIn(lucia, m70), participatesIn(bruno, m70),

participatesIn(bruna, m70), participatesIn(roberta, m70), participatesIn(alfredo, m70), participatesIn(yasmin, m70),

participatesIn(alberto, m70), participatesIn(sofia, m70), participatesIn(jorge, m70), participatesIn(gabriel, m70),

participatesIn(lara, m71), participatesIn(gabriel, m71), participatesIn(marcio, m71), participatesIn(anderson, m71),

participatesIn(alfredo, m71), participatesIn(joaquim, m71), participatesIn(maria, m71), participatesIn(caio, m71),

participatesIn(heloisa, m71), participatesIn(eduarda, m71), participatesIn(carla, m71), participatesIn(diego, m71),

participatesIn(marcelo, m71), participatesIn(manuel, m71), participatesIn(amanda, m71), participatesIn(renata, m71),

participatesIn(renata, m71), participatesIn(guilherme, m71), participatesIn(roberto, m71), participatesIn(roberta, m71),

participatesIn(augusto, m71), participatesIn(julieta, m71), participatesIn(julio, m71), participatesIn(lucia, m71),

participatesIn(bruno, m71), participatesIn(bruna, m71), participatesIn(sonia, m71), participatesIn(jorge, m71),

participatesIn(solange, m71), participatesIn(alberto, m71), participatesIn(luiza, m71), participatesIn(sofia, m71),

participatesIn(yasmin, m71), participatesIn(lara, m72), participatesIn(sofia, m72), participatesIn(leticia, m72),
participatesIn(anderson, m72), participatesIn(alfredo, m72), participatesIn(joaquim, m72), participatesIn(maria, m72),
participatesIn(caio, m72), participatesIn(heloisa, m72), participatesIn(eduarda, m72), participatesIn(carla, m72),
participatesIn(diego, m72), participatesIn(marcelo, m72), participatesIn(manuel, m72), participatesIn(amanda, m72),
participatesIn(thiago, m72), participatesIn(renata, m72), participatesIn(guilherme, m72), participatesIn(roberto, m72),
participatesIn(roberta, m72), participatesIn(augusto, m72), participatesIn(julieta, m72), participatesIn(julio, m72),
participatesIn(lucia, m72), participatesIn(bruno, m72), participatesIn(bruna, m72), participatesIn(sonia, m72),
participatesIn(jorge, m72), participatesIn(solange, m72), participatesIn(alberto, m72), participatesIn(luiza, m72),
participatesIn(gabriel, m72), participatesIn(yasmin, m72),

Table A.13: Background Knowledge

## A.5 Set of Examples

### A.5.1 Positive and Negative Examples

civilMarriage(anderson, solange, luiz, m1), civilMarriage(augusto, julieta, marcio, m2),
civilMarriage(joaquim, maria, leticia, m3), civilMarriage(caio, heloisa, luiz, m4),
civilMarriage(roberta, roberto, marcio, m5), civilMarriage(manuel, eduarda, leticia, m6),
civilMarriage(julio, lucia, luiz, m7), civilMarriage(bruno, bruna, marcio, m8),
civilMarriage(marcelo, sonia, leticia, m9), civilMarriage(marcelo, alfredo, luiz, m10),
civilMarriage(marcelo, jorge, marcio, m11), civilMarriage(marcelo, yasmin, leticia, m12),
civilMarriage(marcelo, alberto, luiz, m13), civilMarriage(marcelo, luiza, marcio, m14),
civilMarriage(marcelo, gabriel, leticia, m15), civilMarriage(marcelo, sofia, luiz, m16),
civilMarriage(carla, sonia, marcio, m17), civilMarriage(carla, alfredo, leticia, m18),
civilMarriage(carla, jorge, luiz, m19), civilMarriage(carla, yasmin, marcio, m20),
civilMarriage(carla, alberto, leticia, m21), civilMarriage(carla, luiza, luiz, m22),
civilMarriage(carla, gabriel, marcio, m23), civilMarriage(carla, sofia, leticia, m24),
civilMarriage(diego, sonia, luiz, m25), civilMarriage(diego, alfredo, marcio, m26),
civilMarriage(diego, jorge, leticia, m27), civilMarriage(diego, yasmin, luiz, m28),

civilMarriage(diego, alberto, marcio, m29), civilMarriage(diego, luiza, leticia, m30), civilMarriage(diego, gabriel, luiz, m31), civilMarriage(diego, sofia, marcio, m32), civilMarriage(amanda, sonia, leticia, m33), civilMarriage(amanda, alfredo, luiz, m34), civilMarriage(amanda, jorge, marcio, m35), civilMarriage(amanda, yasmin, leticia, m36), civilMarriage(amanda, alberto, luiz, m37), civilMarriage(amanda, luiza, marcio, m38), civilMarriage(amanda, gabriel, leticia, m39), civilMarriage(amanda, sofia, luiz, m40), civilMarriage(thiago, sonia, marcio, m41), civilMarriage(thiago, alfredo, leticia, m42), civilMarriage(thiago, jorge, luiz, m43), civilMarriage(thiago, yasmin, marcio, m44), civilMarriage(thiago, alberto, leticia, m45), civilMarriage(thiago, luiza, luiz, m46), civilMarriage(thiago, gabriel, marcio, m47), civilMarriage(thiago, sofia, leticia, m48), civilMarriage(renata, sonia, luiz, m49), civilMarriage(renata, alfredo, marcio, m50), civilMarriage(renata, jorge, leticia, m51), civilMarriage(renata, yasmin, luiz, m52), civilMarriage(renata, alberto, marcio, m53), civilMarriage(renata, luiza, leticia, m54), civilMarriage(renata, gabriel, luiz, m55), civilMarriage(renata, sofia, marcio, m56), civilMarriage(guilherme, sonia, leticia, m57), civilMarriage(guilherme, alfredo, luiz, m58), civilMarriage(guilherme, jorge, marcio, m59), civilMarriage(guilherme, yasmin, leticia, m60), civilMarriage(guilherme, alberto, luiz, m61), civilMarriage(guilherme, luiza, marcio, m62), civilMarriage(guilherme, gabriel, leticia, m63), civilMarriage(guilherme, sofia, luiz, m64), civilMarriage(lara, sonia, marcio, m65), civilMarriage(lara, alfredo, leticia, m66), civilMarriage(lara, jorge, luiz, m67), civilMarriage(lara, yasmin, marcio, m68), civilMarriage(lara, alberto, leticia, m69), civilMarriage(lara, luiza, luiz, m70), civilMarriage(lara, gabriel, marcio, m71), civilMarriage(lara, sofia, leticia, m72), coParticipatesWith(anderson, solange, m1), coParticipatesWith(augusto, julieta, m2), coParticipatesWith(joaquim, maria, m3), coParticipatesWith(caio, heloisa, m4), coParticipatesWith(roberta, roberto, m5), coParticipatesWith(manuel, eduarda, m6), coParticipatesWith(julio, lucia, m7), coParticipatesWith(bruno, bruna, m8), coParticipatesWith(marcelo, sonia, m9), coParticipatesWith(marcelo, alfredo, m10), coParticipatesWith(marcelo, jorge, m11), coParticipatesWith(marcelo, yasmin, m12), coParticipatesWith(marcelo, alberto, m13), coParticipatesWith(marcelo, luiza, m14), coParticipatesWith(marcelo, gabriel, m15), coParticipatesWith(marcelo, sofia, m16), coParticipatesWith(carla, sonia, m17), coParticipatesWith(carla, alfredo, m18), coParticipatesWith(carla, jorge, m19), coParticipatesWith(carla, yasmin, m20), coParticipatesWith(carla, alberto, m21), coParticipatesWith(carla, luiza, m22), coParticipatesWith(carla, gabriel, m23), coParticipatesWith(carla, sofia, m24), coParticipatesWith(diego, sonia, m25), coParticipatesWith(diego, alfredo, m26), coParticipatesWith(diego, jorge, m27), coParticipatesWith(diego, yasmin, m28), coParticipatesWith(diego, alberto, m29), coParticipatesWith(diego, luiza, m30),

coParticipatesWith(diego, gabriel, m31), coParticipatesWith(diego, sofia, m32),
coParticipatesWith(amanda, sonia, m33), coParticipatesWith(amanda, alfredo, m34),
coParticipatesWith(amanda, jorge, m35), coParticipatesWith(amanda, yasmin, m36),
coParticipatesWith(amanda, alberto, m37), coParticipatesWith(amanda, luiza, m38),
coParticipatesWith(amanda, gabriel, m39), coParticipatesWith(amanda, sofia, m40),
coParticipatesWith(thiago, sonia, m41), coParticipatesWith(thiago, alfredo, m42),
coParticipatesWith(thiago, jorge, m43), coParticipatesWith(thiago, yasmin, m44),
coParticipatesWith(thiago, alberto, m45), coParticipatesWith(thiago, luiza, m46),
coParticipatesWith(thiago, gabriel, m47), coParticipatesWith(thiago, sofia, m48),
coParticipatesWith(renata, sonia, m49), coParticipatesWith(renata, alfredo, m50),
coParticipatesWith(renata, jorge, m51), coParticipatesWith(renata, yasmin, m52),
coParticipatesWith(renata, alberto, m53), coParticipatesWith(renata, luiza, m54),
coParticipatesWith(renata, gabriel, m55), coParticipatesWith(renata, sofia, m56),
coParticipatesWith(guilherme, sonia, m57), coParticipatesWith(guilherme, alfredo, m58),
coParticipatesWith(guilherme, jorge, m59), coParticipatesWith(guilherme, yasmin, m60),
coParticipatesWith(guilherme, alberto, m61), coParticipatesWith(guilherme, luiza, m62),
coParticipatesWith(guilherme, gabriel, m63), coParticipatesWith(guilherme, sofia, m64),
coParticipatesWith(lara, sonia, m65), coParticipatesWith(lara, alfredo, m66),
coParticipatesWith(lara, jorge, m67), coParticipatesWith(lara, yasmin, m68),
coParticipatesWith(lara, alberto, m69), coParticipatesWith(lara, luiza, m70),
coParticipatesWith(lara, gabriel, m71), coParticipatesWith(lara, sofia, m72),
coParticipatesWith(m72, solange, m1), coParticipatesWith(augusto, m71, m2),
coParticipatesWith(joaquim, maria, leticia), coParticipatesWith(m70, m69, m4),
coParticipatesWith(m68, roberto, m5), coParticipatesWith(manuel, m67, m6),
coParticipatesWith(julio, lucia, luiz), coParticipatesWith(m66, m65, m8),
coParticipatesWith(m64, sonia, m9), coParticipatesWith(marcelo, m63, m10),
coParticipatesWith(marcelo, jorge, marcio), coParticipatesWith(m62, m61, m12),
coParticipatesWith(m60, alberto, m13), coParticipatesWith(marcelo, m59, m14),
coParticipatesWith(marcelo, gabriel, luiz), coParticipatesWith(m58, m57, m16),
coParticipatesWith(m56, sonia, m17), coParticipatesWith(carla, m55, m18),
coParticipatesWith(carla, jorge, marcio), coParticipatesWith(m54, 53, m20),
coParticipatesWith(m52, alberto, m21), coParticipatesWith(carla, m51, m22),
coParticipatesWith(carla, gabriel, leticia), coParticipatesWith(m50, m49, m24),
coParticipatesWith(m48, sonia, m25), coParticipatesWith(diego, m47, m26),
coParticipatesWith(diego, jorge, luiz), coParticipatesWith(m46, m45, m28),
coParticipatesWith(m44, alberto, m29), coParticipatesWith(diego, m43, m30),
coParticipatesWith(diego, gabriel, marcio), coParticipatesWith(m42, m41, m32),

coParticipatesWith(m40, sonia, m33), coParticipatesWith(amanda, m39, m34), coParticipatesWith(amanda, jorge, leticia), coParticipatesWith(m38, m37, m36), coParticipatesWith(m36, alberto, m37), coParticipatesWith(amanda, m35, m38), coParticipatesWith(amanda, gabriel, luiz), coParticipatesWith(m34, m33, m40), coParticipatesWith(m32, sonia, m41), coParticipatesWith(thiago, m31, m42), coParticipatesWith(thiago, jorge, marcio), coParticipatesWith(m30, m29, m44), coParticipatesWith(m28, alberto, m45), coParticipatesWith(thiago, m27, m46), coParticipatesWith(thiago, gabriel, leticia), coParticipatesWith(m26, m25, m48), coParticipatesWith(m24, sonia, m49), coParticipatesWith(renata, m23, m50), coParticipatesWith(renata, jorge, luiz), coParticipatesWith(m22, m21, m52), coParticipatesWith(m20, alberto, m53), coParticipatesWith(renata, m19, m54), coParticipatesWith(renata, gabriel, marcio), coParticipatesWith(m18, m17, m56), coParticipatesWith(m16, sonia, m57), coParticipatesWith(guilherme, m15, m58), coParticipatesWith(guilherme, jorge, leticia), coParticipatesWith(m14, m13, m60), coParticipatesWith(m12, alberto, m61), coParticipatesWith(guilherme, m11, m62), coParticipatesWith(guilherme, gabriel, luiz), coParticipatesWith(m10, m9, m64), coParticipatesWith(m8, sonia, m65), coParticipatesWith(lara, m7, m66), coParticipatesWith(lara, jorge, marcio), coParticipatesWith(m6, m5, m68), coParticipatesWith(m4, alberto, m69), coParticipatesWith(lara, m3, m70), coParticipatesWith(lara, gabriel, leticia), coParticipatesWith(m2, m1, m72)

Table A.14: Set of Positive and Negative Examples

# Bibliography


[1] Gruber, T. R., ”Toward principles for the design of ontologies used for knowledge sharing”, *International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology*, v. 43, Issue 5-6, pp. 907-928, Nov./Dec. 1995.

[2] Guarino, N., ”Formal ontology, conceptual analysis and knowledge representation”, *International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology*, v. 43, Issue 5-6, pp. 625-640, Nov./Dec. 1995.

[3] Nardi, D., Brachman, R.J., ”An Introduction in Description Logics”. In: *The Description Logic Handbook: Theory, Implementation and Applications*, Cambridge, pp. 1-43, 2003.

[4] Maass, W., Janzen, S., ”A Pattern-based Ontology Building Method for Ambient Environments”. In: *Proceedings of the Worshop on Ontology Patterns WOP 2009, collocated with the 8th International Semantic Web Conference (ISWC-2009)*, v. 516, *CEUR workshop proceedings*, Washington D.C., October 2009.

[5] Völker, J., Rudolph, S., ”Fostering web intelligence by semi-automatic owl ontology refinement”, *Web Intelligence and Intelligent Agent Technology. IEEE/WIC/ACM International Conference*, v. 1, pp. 454-460, Sydney, Dec. 2008.

[6] Gangemi, A., Presutti, V., ”Ontology Design Patterns”. In: *Handbook on Ontologies*, *International Handbooks on Information Systems*, Springer-Verlag, pp. 221-243, 2009.

[7] Presutti, V., Gangemi, A., David, S., et al., *A Library of Ontology Design Patterns: reusable solutions for collaborative design of networked ontologies*, Neon Project, Neon Consortium, 2008.

[8] Falbo, R. A., Barcellos, M. P., Nardi, J. C., et al., "Organizing Ontology Design Patterns as Ontology Pattern Languages". In: *The Semantic Web: Semantics and Big Data*, v. 7882, *Lecture Notes in Computer Science*, Springer-Verlag, pp. 61-75, 2013.

[9] Sure, Y., Staab, S., Studer, R., "Ontology Engineering Methodology". In: *Handbook on Ontologies*, *International Handbooks on Information Systems*, Springer-Verlag, pp. 135-152, 2009.

[10] Leão, F., Revoredo, K., Baião, F., "Um Framework para Refinamento de Ontologias Através de Técnicas de Revisão de Teorias", *Sociedade Brasileira de Computação - Encontro Nacional de Inteligência Artificial*, 2011.

[11] Rudolph, S., Völker, J., Hitzler, P., "Supporting Lexical Ontology Lerning by Relational Exploration". In:*Proceeding ICCS'07 Proceedings of the 15th international conference on Conceptual Structures: Knowledge Architectures for Smart Applications*, pp. 488-491, Sheffield - UK, Jul. 2007.

[12] Cimiano, P., Völker, J., "Text2-Onto: A Framework for Ontology Learning and Data-driven Change Discovery". In: *Proceeding NLDB'05 Proceedings of the 10th international conference on Natural Language Processing and Information Systems*, Springer-Verlag, pp. 227-238, 2005.

[13] Corcho, O., Roussey, C., Blázques, L.M.V., Pérez, I., "Pattern-based OWL Ontology Debugging Guidelines". In: *Proceedings of the Workshop on Ontology Patterns (WOP 2009), collocated with the 8th International Semantic Web Conference (ISWC-2009).*, v. 516, *CEUR workshop proceedings*, CEUR-WS.org, pp. 68-82, Oct. 2009.

[14] Friedrich, G., Shchekotykhin, K., "A General Diagnosis Method for Ontologies". in:*The Semantic Web - ISWc 2005: 4th International Semantic Web Conference, ISWC 2005*, v. 3729, *Lecture Notes in Computer Science*, *Lecture Notes in Computer Science*, pp. 232-246, 2005.

[15] Friedrich, G., Shchekotykhin, K., "Diagnosis discrimination for ontology debugging". In: *Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pp. 991-992, 2010.

[16] Coulet, A., Smail-Tabbone, M., Napoli, A., Devignes, M., "Ontology Refinement through Role Assertion Analysis: example in pharmacogenomics". In:*21st International Workshop on Description Logics*, v. 353, 2008.

[17] Nikitina, N., Rudolph, S., Blohm, S., ”Refining Ontologies by Pattern-Based Completion”, *International Semantic Web Conference*, v. 516, 2009.

[18] Nguyen, T. A. T., Power, R., Piwek, P., Williams, S., ”Measuring the Understandability of Deduction Rules for OWL”. In: *Proceedings of the First International Workshop on Debugging Ontologies and Ontology Mappings - WoDOOM12*, pp. 1-12, 2012.

[19] Nikitina, N., Rudolph, S., Glimm, B., ”Interactive Ontology Revision”, *Web Semantics*, v. 12, pp. 118-130, 2012.

[20] Völker, J., Hitzler, P., Cimiano, P., ” Acquisition of OWL DL Axioms from Lexical Resources”. In: *The Semantic Web: Research and Applications*, v. 4519, *Lecture Notes in Computer Science*, Springer-Verlag, pp. 670-685, 2007.

[21] Wrobel, S., ”First Order Theory Refinement”. In: *Advanced in ILP*, pp. 14-33, Amsterdam, 1995.

[22] Vasconcellos, Wander, Revoredo Kate, Baião, F., ”Refinamento de Ontologias utilizando Revisão de Teoria e Padrões de Projeto de Ontologias”, *SBSI: Workshop de Teses e Dissertações em Sistemas de Informação*, 2013.

[23] Vasconcellos, W., Revoredo, K., Baião, F., ”Improving Ontology Refinement through Theory Revision and Ontology Design Patterns”, *Sociedade Brasileira de Computação - Encontro Nacional de Inteligência Artificial e Computacional*, 2013.

[24] Richards, B. L., Mooney, R. J., ”Automated Refinement of First-Order Horn-Clause Domain Theories”, *Machine Learning*, v. 19, Issue 2, pp. 95-131, May 1995.

[25] Völker, J., Rudolph, S., ”Lexico-Logical Acquisition of OWL DL Axioms”. In: *Formal Concept Analysis: 6th International Conference, ICFCA 2008, Montreal, Canada, Frebuary 25-28, 2008. Proceedings*, v. 4903, *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 62-77, 2008.

[26] Horridge, M., Aranguren, M. E., Mortensen, J., et al., ”Ontology Design Pattern Language Expressivity Requirements”. In: *WOP*, v. 929, *CEUR Workshop Proceedings*, CEUR-WS.org, 2012.

[27] Gangemi, A., ”Ontology Design Patterns for Semantic Web Content”. In: *The Semantic Web - ISWC 2005*, v. 3729, *Lecture Notes in Computer Science*, Springer-Verlag, pp. 262-276, 2005.

[28] Sommer, E., ”Theory Restructuring: Coarse-grained Integration of Strategies for Induction & Maintenance of Knowledge Bases”. In: *Proceedings of the Third International Conference on Multistrategy Learning (MLS-96)*, pp. 177-190, Palo Alto, May 1996.

[29] Duboc, A.L., Paes, A., Zaverucha, G., ”Using the Bottom Clause and Mode Declarations in FOL Theory Revision from Examples”, *Machine Learning*, v.76, pp. 73-109, 2009.

[30] Mitchel, Tom M., 1997, *Machine Learning*. 1 ed. New York, McGraw-Hill, 1997.

[31] Muggleton, Stephen, 1991, ”Inductive Logic Programming”. New Generation Computing 8 (4), pp. 295-318, 1991.

[32] Muggleton, Stephen, Raedt, Luc de, 1994, ”Inductive Logic Programming: Theory and Methods”, *Journal of Logic Programming*, v. 19, pp. 629-679, 1994.

[33] Witten, I. H., Frank, E., Hall, M.A., *Data Mining: Pratical Machine Learning Tools and Techniques*. 3 ed. Elsevier, 2011.

[34] Baader, F., Calvanese, D., McGuinness, D.L., et al, *The Description Logic Handbook: Theory, Implementation and Applications*, Cambridge University Press, 2003.