

ONTOLOGY BASED MODEL FOR THE ITU-T RECOMMENDATION G.805: TOWARDS THE SELF-MANAGEMENT OF TRANSPORT NETWORKS

Maxwell E. Monteiro¹, Anilton Salles Garcia¹, Pedro Paulo F. Barcelos¹, and Giancarlo Guizzardi²

- 1- LabTel – Telecommunications Research Laboratory, Electrical Engineering Department, Federal University of Espírito Santo, Vitória, ES, Brazil
anilton@inf.ufes.br, maxmonte.vix@gmail.com,
pedropaulofb@gmail.com
- 2- Department of Informatics, Federal University of Espírito Santo, Vitória-ES, Brazil
gguizzardi@inf.ufes.br

ABSTRACT

This paper describes an ontology model (conceptual model) based on the ITU-T Recommendation G.805 for transport network architectures. The goal is to take a step forward to the autonomic applications for transport network operation and management. Besides the conceptual model description and explanation, including its main elements (classes, properties and constraints), this paper also includes the description of the G.805 recommendation main concepts characteristics. Furthermore, the paper elaborates on the advantages of using an ontological model, as well as the technologies used for ontology conceptual modelling and ontology implementation. Finally, this paper presents a case study using a simple OTN unidirectional network, in which automated inferences could be verified. In conclusion, the methodology and reference conceptual model of transport networks presented here constitutes a contribution towards autonomic network applications.

KEYWORDS

Transport Networks, ITU-T G.805, Ontology Model, Computational Ontology, Self-management

1. INTRODUCTION

The topics of self-management and autonomic network systems have gained interest in many conferences and specialized publications. There are several approaches to guide the construction of this class of system: Control Theory, Logic, Bio-algorithms, etc; but what can be noticed is that the key point for autonomy and flexibility is the knowledge and Information Model behind those approaches.

In Multi-technology Transport Networks, in which different technologies can potentially work together, autonomous network operation and management become a challenge. For this class of problem, a consistent Knowledge and Information Model is needed to represent a transport network from its abstract view up to its technological specific (drill down) view.

The ITU-T Recommendation G.805 [1], named “*Generic functional architecture of transport networks*”, offers a functional and structural model to represent, in an abstract way, the architecture of a technology independent transport networks. Its main objective is to establish concepts, definitions and a visual representation that could be used as a base model for any specific transport technology. Examples include ATM (Asynchronous Transfer Mode), SDH (Synchronous Digital Hierarchy), PDH (Plesyochronous Digital Hierarchy) and OTN (Optical

Transport Networks). Besides definition and concepts, the ITU-T Recommendation G.805 presents an abstract relationship between different technologies working together, the so-called *Client-Server Technological Layer Relationship*. In the recommendation, a visual language, created to support human interpretation, shows the main concepts and definitions as symbols, giving to the reader a visual tool to describe a transport network.

The ITU-T G.805 and its co-related recommendations (for example, ITU-T G.872 [2] that deals specifically with OTN) are well known in the International Telecommunication Community. The G.805 has been used for years by component developers, vendors and telecommunications service operators. This demonstrates its relevance and coherence in supporting interoperability and new developments in the telecommunication industry. For all those features, the ITU-T G805 can be taken as a reference starting point for the development of a Knowledge and Information Model to be used for network autonomic system, especially in the Network Operation and Management area.

In spite of all those positive features, the ITU-T G.805 is written in natural language, to be read and understood by humans. However, as it is often the case with natural language specifications, it contains many ambiguous and obscure declarations that constitute an obstacle even to human-to-human communication. There are examples of concepts that are too abstract in the recommendation and their real meaning only become clear when the reader observes some samples written in the visual language. In other cases, some visual representations conflict with the textual description. Overall, there is a significant quantity of implicit and/or ambiguous concepts and definitions in ITU-T G.805. As consequence, the successful interpretation and understanding of ITU-T Recommendation G.805 depends heavily on the reader's experience and familiarity with the subject of transport networks. This complicates the task of extracting a consistent knowledge and information model from it, especially if the only tools at hand are conventional Software Engineering Modelling Languages (UML, ER, etc).

In order to support the process of building autonomic applications, a consistent, unambiguous and formal (well-described and formulated) information model is crucial. That formal model should be able of capture and describe terms, definitions, concepts and relationship of a subject domain. The research work presented in this paper has as its main objective to propose an initial version of a Conceptual Model for Abstract Transport Networks Description, taking as its primary source of domain knowledge the ITU-T Recommendation G.805. The model has been designed using an ontologically well-founded conceptual modelling language. This feature of the language allows to systematically developing a precise specification of Transport Networks Information Domain reducing potential ambiguities and exposing implicit knowledge.

1.1. The Solution Approach Proposed

After the advancement of Semantic Web vision and its consequent increasing popularity, the computational ontology paradigm (or simply computational ontology) has been used in many human knowledge areas to support expert computer applications in domains such as electrocardiogram interpretation [3], Grid Computing Configuration [4] and Configuration Management for Information Technology Infrastructure Library [5]. For a general introduction to the field, the reader can refer to, for example, [11]. The OWL (Web Ontology Language) [6] has gained recent popularity as one the most used representation languages for modern computational applications that support automated reasoning. One of the reasons for this popularity is due to its role in affording the combination of web development with formal methods (descriptive logic, horn clauses, etc). However, a language such as OWL, designed to be handled by computers on the Internet, may not have sufficient expressive power to thoroughly represent an information Domain. Its use as a primary conceptual modelling language can be inappropriate and, in some cases, can render the produced specifications

seriously incomplete. This paper uses the approach proposed by Guizzardi [7], where the information and knowledge modelling process is divided into two steps:

- **Domain Conceptual Modelling** – In this first step, domain knowledge and information are formally described using an ontologically well-founded conceptual modelling language and having conceptual clarity and representation adequacy as the main concerns. In this paper, in order to produce the conceptual models for the ITU-T G.805 recommendation, we have used the ontologically well-founded conceptual modelling language OntoUML proposed in [15] extended with first-order logic domain constraints;
- **Computational Implementation** – where the conceptual model is translated into (possibly many) implementation artefacts that can, for example, be interpreted and processed by machines. Each of these mappings aim at maximizing a specific set of non-functional requirements, which are relevant for some specific application purposes.

A more elaborated view of this modelling process is presented in section 2.

1.2 Related Works

In the best effort of the authors, only one work could be found which proposes the creation of structural conceptual models to describe telecommunications networks with the goal of increasing flexibility and autonomy of network software. The Network Description Language (NDL) [8] has been developed using RDF (Resource Description Framework) and RDFS (Resource Description Framework Schema) [9]. The objective is to address the problems of model and concepts interoperability in a telecommunication networks resource description. NDL has been used primarily for inter-domain routing problem.

Its current development level allows it to be used to create tools where network resources description and location are needed. However, as explained by the authors, NDL does not fulfil the ITU-T Recommendation G.805 concepts and definitions. Furthermore, its model implementation using RDF and RDFS imposes restrictions on its expressiveness and may cause restrictions for its use in other applications. As we have previously discussed, formalization techniques with no support to well-founded conceptual modelling principles can render models that lack relevant information and may be deficient in terms of expressivity, clarity and reusability.

1.3 Our Contribution

Telecommunications Networks Conceptual Models are not easily found in research community. An open, extensible and reusable conceptual model as well as a consistent implementation of it would be a very interesting step forward to autonomic systems for networks operation and management.

The main contribution of this paper is to present an initial Transport Networks Conceptual Model, developed using a modeling language based on a Foundational Ontology. The model, which follows the ITU-T Recommendation G.805, could be reused and extended as a baseline for the definitions of relationships and concepts of the Transport Networks Domain. Another contribution is the proposed conceptual model implementation written in computational ontology language (OWL-DL [6]) and extended by rules (using the SWRL extension to OWL - Semantic Web Rule Language [10]). The latter enables the creation of computer applications that can support automated reasoning and, hence, the derivation of information based on the original model.

1.4 – Text Organization

Section 2 briefly presents the technical background on the ITU-T Recommendation G.805 and Ontology Engineering methodology. Section 3 presents the ITU-T G.805 Conceptual Model through diagrams and explanation of some key points. In order to do that, we have used OntoUML diagrams extended by a set of First Order Logical Rules. Section 4 is dedicated to the conceptual model implementation, using OWL-DL and SWRL. Besides that, section 4 presents some results of machine inference on the implemented model. Finally, in section 5, conclusions and future works are presented.

2. TECHNICAL BACKGROUND

2.1 ITU-T Recommendation G.805

In its chapter 5, the ITU-T Recommendation G.805 defines a generic model to describe the transport networks architecture. This functional and structural model proposed in ITU-T G.805 provides a high level of abstraction for the basic elements in a network and defines relevant concepts to simplify network descriptions: Partitioning (some elements can be part of others or be composed of others from the same kind) and Layering (each technology is inside a Layer and different aspects of a complex network can be viewed from different Layers). Furthermore, the recommendation defines the client-server relationship between vertically adjacent Layers.

The independence of the architecture model from specific technologies is powerful relevant feature. The model provides means to describe different kinds of complex transport network. This feature turns the G.805 the basis for other technology specific recommendations, such as the ITU-T Recommendation G.872, named “Architecture of optical transport networks” (OTN). The ITU-T G.805 also contains, in addition to a textual (in natural language) description of the main concepts and its relationships, a visual language to represent the same concepts and relationships and to support a better understanding of the document.

The basic and abstract architectural components described in the G.805 recommendation have three types of directionality: bidirectional, source unidirectional (origin) or sink unidirectional (destination). The unidirectional network always has a unique information flow direction: components from source end transfer network information to the sink end, passing through the various Layers that compose the Transport Network. A bidirectional network is composed of bidirectional architectural components which have a pair of sink and source components co-located. In this case there are transmission and reception of information in both directions (ends A and B) of the network.

Many definitions and concepts in ITU-T Recommendation G.805 are recursive. They keep the model abstract and valid for the majority of specific technology used in transport networks. Examples of that feature are the orthogonal concepts of Network Partitioning (horizontal recursive definition) and Layering (vertical recursive definition). This vertical view, the Layering concept, organizes a transport network in adjacent layers, separating different technologies and network protocols. However, each layer is described using the same basic and abstract elements, making it easier to model multi-technology networks. The horizontal view, the Partitioning concept, organizes the topological elements on a network layer. Partitioning is important to describe routing aspects, administrative domain boundaries and the sub-network (a recursive definition for a not well-known network, e.g. cloud network).

As mentioned before, the flow of information between the two ends of the network (source and sink), is performed through adjacent layers up to the real transmission at the lowest layer. These adjacent layers have a client/server relationship where a lower-level layer (server) provides the transport services to the higher-level layer (client). An example of client/server relationship occurs between the OCh and OMS layers in optical transport networks (OTN). It is important to

observe that client/server relationship is not dependent on information flow directionality (uni or bi-directional). It only depends on network layer organization (technology and protocols).

Besides Partitioning and Layering, other important definitions are: Transport Processing Functions (TPF) and Reference Points (RP). The TPF are blocks which process information that pass through them by their input and output ports. There are two types of transport processing functions: Trail Termination Function (TTF) and Adaptation Function (AF). The RP represents a binding between input and output ports of different instances of TPF. There are three types of RPs: AP (Access Point), CP (Connection Point) and TCP (Termination Connection Point).

An example of abstract transport network using the ITU-T G.805 visual language is presented in Figure 1.

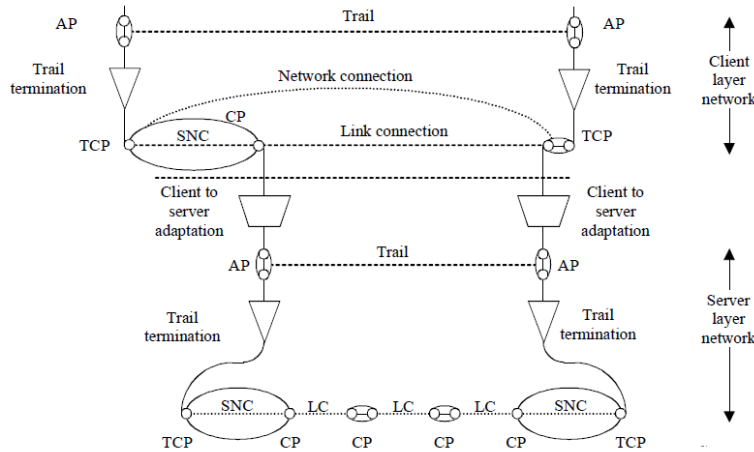


Figure 1. Example of Transport Network [1]

2.2. The Ontology Construction Methodology

The methodology for Ontology construction used in this paper is based on Guizzardi's [7] approach and it has two steps:

- Domain Conceptual Modelling; and
- Computational Implementation.

Domain Conceptual Modelling is a methodological step where the main concepts and definitions of a certain domain are captured using a conceptual modeling language designed with the specific purpose of maximizing ontological expressiveness and conceptual clarity. In other words, that language must be able to capture concepts and relationships in a way which attempts at maximizing representation adequacy regardless of computational restrictions and commitments. To do that, such language must contain modeling primitives based on Foundational Ontologies, aimed to describe the relations between the most basic elements of any domain [7].

The Conceptual Modelling of ITU-T Recommendation G.805 was built using the conceptual modeling language named OntoUML proposed by Guizzardi [15] extended with domain constraints captured as first-order logical rules. It is a language based on UML grammatical primitives, but it was expanded to reflect ontological distinctions described in UFO (Unified Foundational Ontology) [16]. UFO is a Foundational Ontology that guarantees basic primitive concepts over which domain-specific ontologies can be built.

OntoUML comprises modeling primitive to capture different sorts of ontological distinctions among the categories of formal relationships (mediation, derivation, characterization, etc.) and types (kinds, roles, relators, etc.) that helps the modeler in the construction of ontologically correct and consistent models. OntoUML has been successfully applied for the construction of domain reference models in several different domain (e.g., [3, 17]).

Computational Implementation, the second step of the methodology, is the translation of a conceptual model (from first step) to a codification language focused on computational concerns (e.g., decidability and computational tractability). In this work, the combination of OWL-DL and SWRL has been chosen to implement the original conceptual model.

This methodology has an important advantage which is to offer a long term model (Conceptual Model), which can be used to generate different implementations, depending on specific project characteristics.

3. CONCEPTUAL MODELLING OF AN ITU-T G.805 TRANSPORT NETWORK

3.1. The Scope of the Model

The ITU-T Recommendation G.805 model presented in this paper should be regarded as a work in progress. The current scope includes modelling the vision in layers and the concepts and definitions directly involved - establishing the client/server relationship - as well as concepts of point-to-point information transfer in a network, involving Trails, Connections and its subtypes.

Six different models were generated: Main Model, for the connection between layers and components; the Source Model, for the source side of the network, where TTFs and AFs connections are represented; a third model for the network destination, called Sink Model, similar to the source side; the fourth model representing the reference points (Reference Points Model), the fifth model representing the transport entities (Transport Entities Model); and, finally, a sixth model, named Horizontal Model, with entities and relationships that represents horizontal transfer of information.

3.2. The Main Descriptive Model

The Main Model, shown in Figure 2, has layers (type Layer) Trail Termination Functions (type TTF) and Adaptation Functions (type AF) as main types.

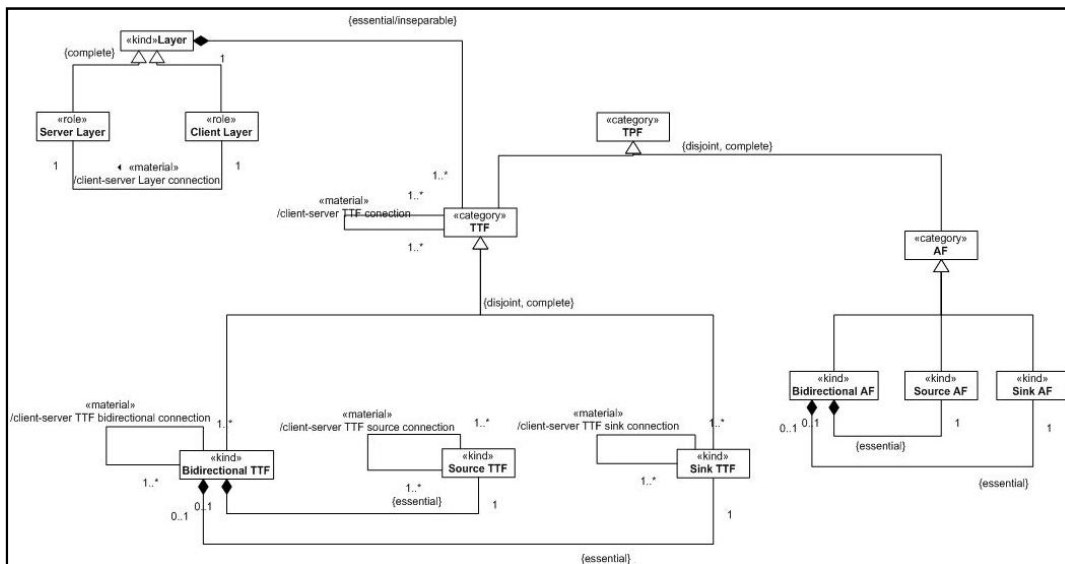


Figure 2. Main Model

Other specialized types of the Main Model have also been modeled. For example, the roles Server Layer and Client Layer capture different roles a layer can play. As it can be observed, these types are not disjoint, i.e., the very same layer can play the role of server with a layer above, and the role of client with layer below.

The TPF class (Transport Processing Function) can be further specialized in two different subtypes, TTF or AF. These two types are, however, disjoint, i.e., they cannot have instances in common.

Both TTF and AF types (subtypes of TPF) can be specialized into three disjoint types, depending on their direction: source, sink and bidirectional. It can also be observed that the bidirectional component is composed of exactly one source component and one sink component. We also highlight the composition of the layers by TTFs. According to ITU-T Recommendation G.805, TTFs are essential parts of the layers, i.e., there are one or more TTF instances as part of a layer, and a layer cannot exist without that specific collection of TTFs.

The client-server relationship between layers is modeled as a *client-server Layer connection*. The cardinality of this relationship is always 1 to 1 and its directionality is from Client Layer to Server Layer. Another relationship modelled is between TTFs of different layers, one from a client layer and the other from a server layer, the *client-server TTF connection*. This is not a physical relationship and it is not described by the recommendation, but it is important for the defining integrity and derivation rules in the model. Depending on the directionality of the related TTFs, this relationship can be of three types: source, sink or bidirectional.

3.3. The Source and Sink Transport Processing Function Descriptive Model

The Source and Sink Transport Processing Function (TPF) models are similar, having the same structure. They differ, however, in terms of directionality and in terms of the kind of reference point connections that constitute them. In the case of the Source TTF, we have a Source TTF Output and a Source AF Input connected by a Source TCP, and the Source TTF Input connected to a Source AF Input by a Source AP. In contrast, in the case of a Sink TTF, the Sink AP connects the (Sink) TTF Output with the (Sink) AF Input; the Sink TCP then connects the TTF Input with the AF Output. The TPF source model is shown in Figure 3 below.

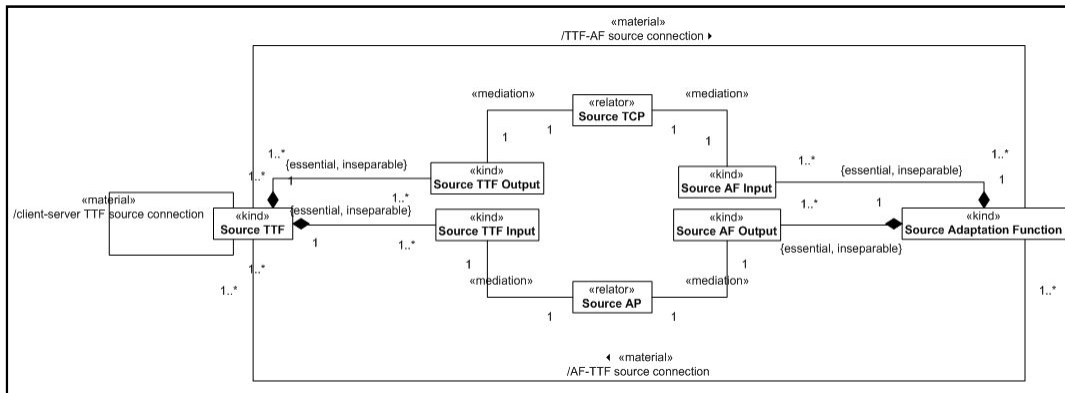


Figure 3. TPF Source Model

The trail termination functions (TTF) are entities placed within layers according to the Recommendation. The adaptation functions (AF) are transport processing functions, which are located between layers and have the function of adapting the information that is transferred

between adjacent layers. For the source side of the network, the *TTF-AF source connection* relationship represents the physical connection between a TTF and an AF. This relationship occurs when an output of TTF is physically connected via a TCP (Termination Connection Point) to an AF input.

A TCP is created when a connection between these two components occurs and it represents a reference point to identify this connection – it is not a physical component, but it has indicative and representative functions. The TCP is destroyed when the connection is broken, ending the *TTF-AF source connection* relationship. The *AF-TTF source connection* relationship uses the same idea, but in this case, it is the connection of an output of an AF with an input of a TTF (in the case of the source side of the network) and the reference point is an AP (Access Point).

3.4. The Reference Points and Transport Entities Descriptive Model

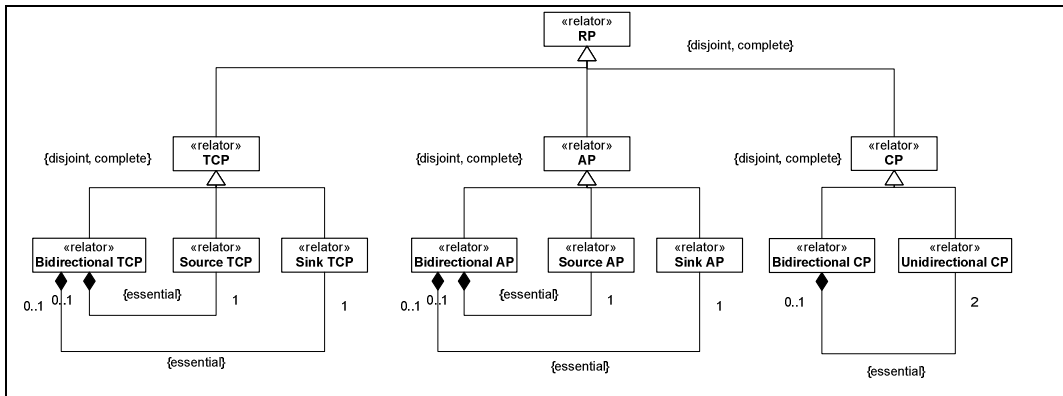


Figure 4. Reference Points Model

The model presented on Figure 4 depicts the possible types of reference points. Moreover, it represents that Bidirectional TCPs as well as Bidirectional APs are composed of a source-sink pair of, respectively, TCPs and APs that are co-located. Unlike these two concepts, a Bidirectional CP is composed of two Unidirectional CPs co-located, with no directionality differentiation.

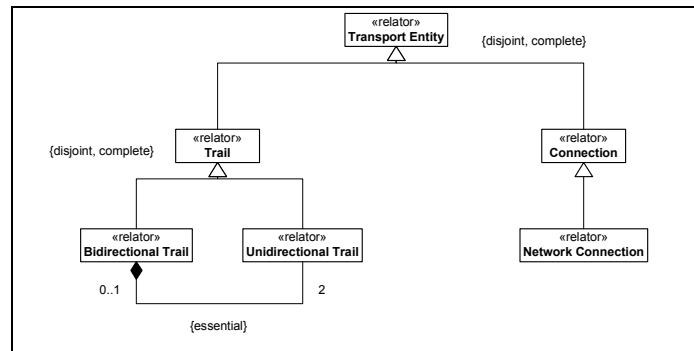


Figure 5. Transport Entities Model

In Figure 5, the Model of Transport Entities classifies them according to their type (Trails or Connections). It is important to note that Network Connection is not the only type of Connection existing in the ITU-T Recommendation G.805. There are also link connections (LCs) and the subnetwork connections (SNC). The latter concepts have not been contemplated in the current version of the conceptual model.

3.5. The Network Horizontal Relations Descriptive Model

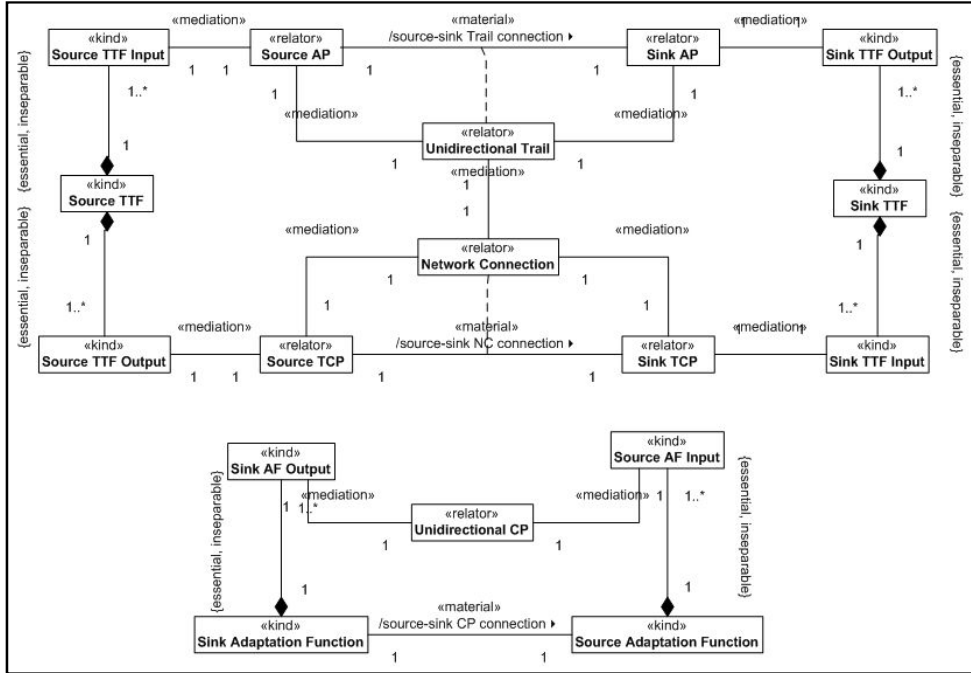


Figure 6. Horizontal Model

The main entities modeled here (Figure 6) are Unidirectional Trail, Network Connection and Unidirectional CP. A Unidirectional Trail is formed by the association of a Source AP with a Sink AP, relating them via the *source-sink Trail connection*. This Unidirectional Trail represents the transfer of information monitored between two APs. To make the Trail possible, the network must have a Network Connection binding its TCPs at the same network layer. Thus, a Unidirectional Trail is existentially dependent on a Network Connection. A Network Connection, in turn, represents the transfer of information between TCPs and requires the existence of a (generic) Trail. The Network Connection is modeled as a *relator* (OntoUML primitive), linking two TCPs (of source and sink types) with a *source-sink NC connection* relationship.

A Unidirectional CP is important for the network since it completes the connection between the network source and sink sides in some situations where there is no direct (continuous) link connection between TCPs of a Network Connection. Physically, it relates either: the output of a Sink AF with the input of a Source AF; a Source TCP and a Sink AF output; or a Source AF input and a Sink TCP. In any case, its role is to ensure the transmission of information. We have named the relationship between AFs related by a CP *source-sink CP connection*.

3.6. First Order Logic Rules

In order to formally represent a number of integrity and derivation rules, which are part of this domain, we have extended the OntoUML models with first order logic rules. As the integrity rules model, for example, the need of the components of the Bidirectional TCPs being co-located (the same goes also for the APs, TTFs and AFs). In contrast, derivation rules exist to enable inferences of concepts and relationships from the explicitly modeled elements. Sixteen general rules have been used to extend the model. Due to limited space, only some of these rules are shown as examples.

The first rule that can be cited as an example is the one that derives client-server relationships between layers (*client-server Layer connection*) when the TTFs of these layers are related (*client-server TTF connection*). Rule 1 is the formal description of it:

$$\forall x, y \text{ isPartOf}(x,a) \wedge \text{isPartOf}(y,b) \wedge \text{client-server TTF connection}(x,y) \rightarrow \text{client-server Layer connection}(a,b) \quad \text{[Rule 1]}$$

This rule states that for any x and y TTFs which are, respectively, part of a and b Layers and that x has *client-server TTF connection* with y, it is known that Layers a and b have a client-server relationship, being the Client Layer and b the Server Layer. Figure 7.A represents this rule.

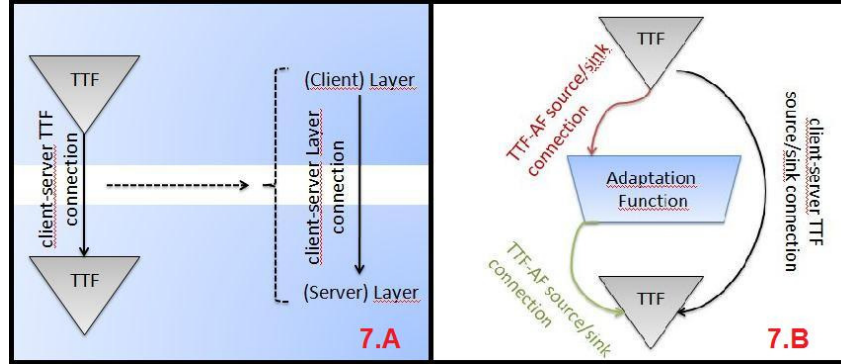


Figure 7. Vertical Rules Examples

By knowing the relationship between TTFs of two different layers, it is also possible to infer that this occurs through physical connections between these TTFs and an AF that adapts the exchanged information between them. The inverse of this is also valid. By knowing that an AF relates two TTFs, it can be inferred that there is a *client-server TTF connection* relationship between the TTFs. The formal description of Rule 2, for the source side of the network, is:

$$\forall x, y, z \quad \text{TTF-AF source connection}(x,y) \wedge \text{AF-TTF source connection}(y,z) \leftrightarrow \text{client-server TTF source connection}(x,z) \quad \text{[Rule 2]}$$

It is important to note that these rules are interrelated. For example, in the rule described above (Figure 7.B), it can be inferred that the TTFs are related through *client-server TTF source connection* and, from the first rule described in this subsection is also possible to infer that these two TTFs are in adjacent layers with a client-server relationship.

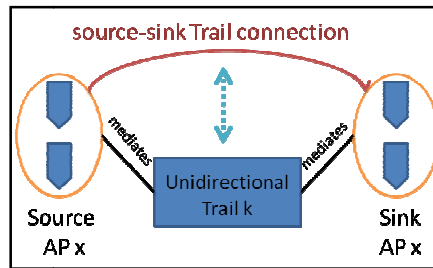


Figure 8. Horizontal Rules Examples - Trail

Figure 8 presents the relationship *source-sink Trail connection* that comes up when there is a Trail between source and sink APs at the same network layer, formalized by Rule 3.

$$\forall x, y, k \text{ Sink AP}(y) \wedge \text{Source AP}(x) \wedge \text{Unidirectional Trail}(k) \wedge \text{mediates}(k, x) \wedge \text{mediates}(k, y) \rightarrow \text{source-sink Trail connection}(x, y) \quad \text{[Rule 3]}$$

Many other rules were created to formalize implicit aspects of the ITU-T G.805. Some examples are: Subtypes of client-server TTF connection, Relationships of Bidirectional TTF components, TCP relation with TTF-AF, AP relation with AF-TTF, Bidirectional TTFs-AFs and AF-TTF Complementary Relationships, Existence and Composition of Bidirectional APs, TCPs and Trails, Trail and NC Relation with Horizontal Relationships, Ensuring Relations with Horizontal Components of the Same Layer, Inference of Horizontal Relationship by knowing the network connection, and Inference of Horizontal Relationships by Client-Server TTFs.

It can be noted that the rules here formalized are not clearly described in the ITU-T Recommendation G.805. That is especially important when we think about a multi-technology transmission network, where different transport technologies can be combined in different positions with client-server relationship. Thus, there must be a clear way to define which technology serves and which one is served within a given transmission network.

4. CONCEPTUAL MODEL IMPLEMENTATION

As an example of how a well structured OWL specification can help us to infer useful information from a network, we present here the implementation of a simple unidirectional point-to-point OTN network containing 45 individuals (instances of classes). These individuals and the declared knowledge of the network can be seen in Figure 9 below. The OWL file generated in this step uses 37 classes, 15 relationships, almost 100 axioms and 23 SWRL rules.

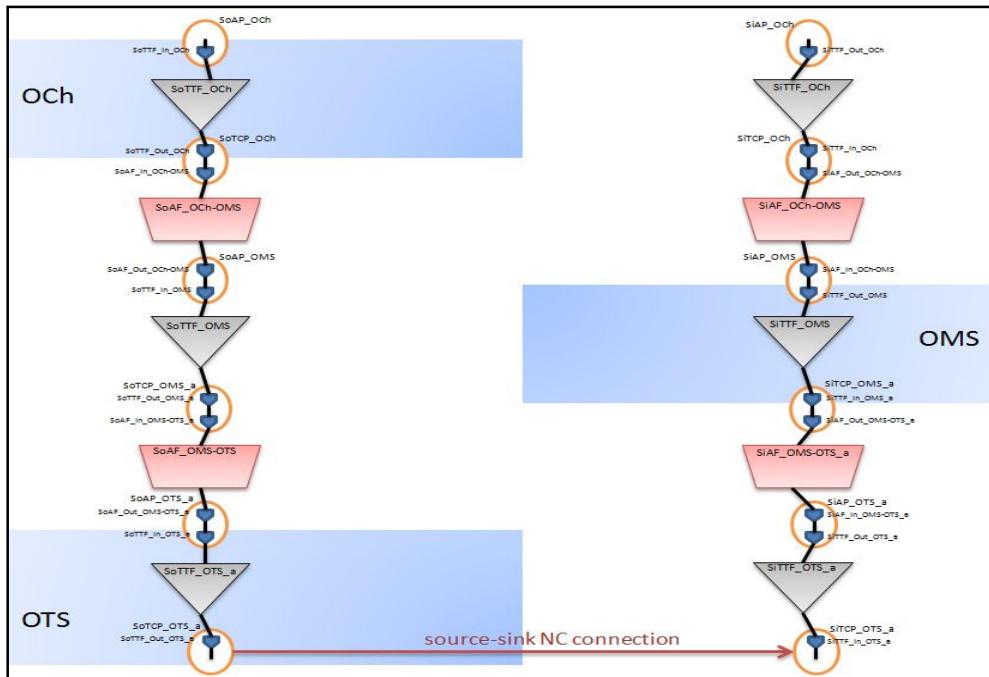


Figure 9. Declared OTN Network

In the subtitle of Figure 9, the term “declared” means that the elements and relations in this figure constitute the knowledge which has been explicitly asserted in the model. Our intention with this example is to demonstrate how new information can be derived from this explicitly declared subset by using an OWL specification and its associated inference rules.

Figure 9 uses for component representation the same visual language available in the ITU-T Recommendation G.805: gray triangles are TTFs, red trapezoids are AFs and the yellow circles are Reference Points (APs or TCPs), the latter containing component Inputs and Outputs inside. All named components in the network were implemented as individuals in the OWL file. We want to emphasize that OCh, OMS and OTS are individuals from the Layer class; no client-server relation were explicitly declared about them. The only information asserted about the classes are that the individual SoTTF_OCh (Source TTF) is part of the OCh layer, SiTTF_OMS (Sink TTF) is part of layer OMS and SoTTF_OTS is part of layer OTS. The physical path from the source side of the network to the sink side is guaranteed by the *source-sink NC connection* in this layer.

Once we know all physical connections in a transport network, what additional information can be obtained from it? Invoking the Pellet *reasoner*, the answer can be found in Figure 10.

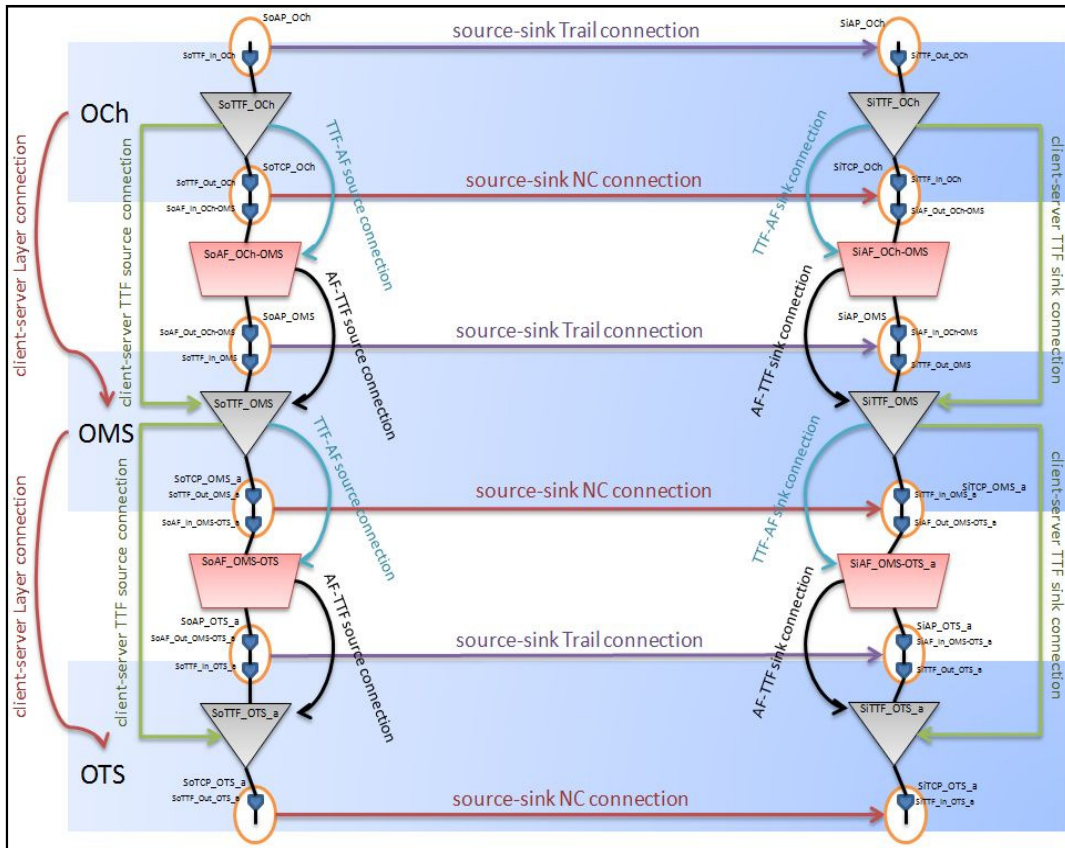


Figure 10. Inferred OTN Network

All colored relations in this figure were automatically derived as a result of the reasoning process. Vertically in the model we can find characteristic information transmission relations between components in a physical path. The *client-server TTF source/sink connection* (in green) between TTFs contained in a client layer to TTFs contained in a server layer and that are in a physical path and, as the *TTF-AF source/sink connection* (in blue) and *AF-TTF source/sink connection* (in black) are representation of how information flows in a layering view inside a network. All *client-server TTF source/sink connection* relationships in this network were inferred by Rule 2 (Figure 7.B), implemented as an SWRL rule.

In the layering view, another important inference was about the client-server relationship of layers. On the very left of Figure 10, we can see that *client-server Layer connections* were inferred between the OTN layers. On its left side, Figure 11 (a Protégé screenshot) shows the asserted information about the layers and on its right side, the *reasoner's* automatic classification results.

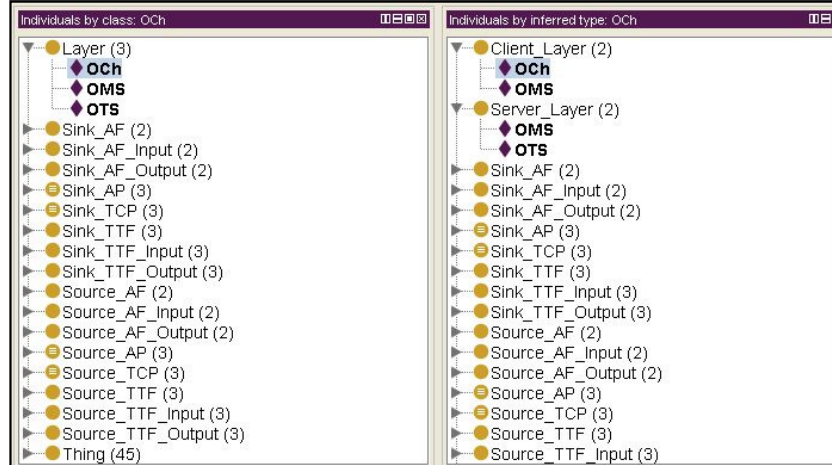


Figure 11. Declared and Inferred Classification of the Layers

This inference was possible on account of the use of Rule 1 by relating TTFs through the *client-server TTF connection* which entails a relationship between Layers. Knowing that *client-server Layer connections* property has as domain the class Client Layer and as range the class Server Layer, OCh, OMS and OTS were automatically classified as these. OCh is a client layer (of OMS layer), OMS is both client (of OTS) and server (of OCh) and OTS is a server layer. Always bearing in mind that the Layer subtypes in the model are not disjoint from each other (as discussed in section 3.1), OMS being classified both an instance of Client Layer and Server Layer is not an inconsistency in this model.

Other important information that was inferred in the model was the horizontal relationships depicted in figure 11. Once more, by using SWRL rules, it was possible to infer from the asserted network that all reference points are related, representing point-to-point transfer of information. Those relationships (*source-sink Trail connection*, in purple, and *source-sink NC connection*, in red) represent the existence of a link, between the related reference points, with transfer of information.

By automatically monitoring the relationships described here, it is possible to see if a network is normally functioning. By knowing that a relationship no longer exists, we can easily see the point where a problem occurred and act on it so as to solve the problem. As an extension to the work presented here, we envisage a scenario in which automatic component discovery processes can upload information (individuals) to this specification and the knowledge obtained from this process can then feed management systems.

5. CONCLUSIONS

Autonomic network applications are an important topic of academic research. It can be noted that flexibility and autonomy depend on some key features in this networks, and that the knowledge and Information Model is one of the most relevant components enabling such features.

The ITU-T Recommendation G.805 has a significant contribution in the description of a Generic Functional Architecture for Transport Networks in a technology independent way. However, G.805 was made only for human reading and can benefit from a sound interpretation with an associated formalization.

This paper has presented an initial conceptual model for the ITU-T Recommendation G.805, by employing an ontologically well-founded conceptual modeling language. The resulting model adds precision and clarity to the interpretation of Transport Networks concepts and definitions. Furthermore, the paper reports on an OWL /SWRL implementation of this conceptual model.

A validation application scenario, presented in Section 4, was performed over a simple unidirectional OTN network. The results have shown that a machine (the *reasoner*) automatically conduct correct inferences over the model. This constitutes a step forward towards the goal of building autonomic applications.

We envisage that the ontology-based model proposed in this paper can be used in applications that are capable, for example, of:

- Evaluating a transport network by its functional components and their interconnections, assisting in the setting of these;
- Planning and simulating transport networks;
- Correlating alarms and indicators monitored within a multi-technology (layers with different technologies) network;
- Establishing and monitoring the end-to-end level of service of transport service, including dependencies, the capacity and function of each network element that affects that service;

As previously discussed, the model proposed here constitutes a preliminary effort and, hence, it needs to be extended to capture other aspects of ITU-T G.805. These aspects include: Multiplexing, Protection and Monitoring. A natural evolution of this work is to extend the model to represent the ITU-T Recommendation G.872 (Architecture of Optical Transport Networks) as well as the modeling of the ITU-T G.798 and the G.806 recommendations.

ACKNOWLEDGEMENTS

The authors would like to thank Padtec S.A and CNPq for sponsoring this paper. The work of the last authors has been supported by CNPq (grant number 309382/2008-4) and FAPES (grant number 45444080/09)

REFERENCES

- [1] ITU-T. G.805: Generic functional architecture of transport networks. International Telecommunication Union. [Online] March 10th 2000. [Accessed on: February 25, 2009]. <http://www.itu.int/rec/T-REC-G.805/en>.
- [2] ITU-T. G.872: Architecture of optical transport networks. International Telecommunication Union. [Online] November 29th 2001. [Accessed on: February 25, 2009]. <http://www.itu.int/rec/T-REC-G.872/en>.
- [3] Gonçalves, B. N.; Zamborlini, V.; Guizzardi, G. An Ontological Analysis of the Electrocardiogram. *Electronic Journal of Communication, Information and Innovation in Health*, 2009.
- [4] Pernas, Ana M. e Dantas, M. A. R. *Ontologies Applied to Description of Resources in Grid Environments*. 2004. [In Portuguese].

- [5] Baiôco et al. IT Service Management and Governance - Modeling an ITSM Configuration Process: a Foundational Ontology Approach. IFIP/IEEE Intl. Symposium on Integrated Network Management— Workshops 978-1-4244-3924-9. 2009
- [6] Patel-Schneider, P.; Hayes, P.; Horrocks, I. OWL Web Ontology Language Semantics and Abstract Syntax. W3C Recommendation. [Online] 2004. <http://www.w3.org/TR/owl-semantics/>.
- [7] Guizzardi, G., Halpin, T. **Ontological Foundations for Conceptual Modeling**. Applied Ontology. , v.3, p.91 - 110, 2008.
- [8] Research Group System and Network Engineering. Network Description Language. [Online] <http://www.science.uva.nl/research/sne/ndl/>.
- [9] W3C. Resource Description Framework (RDF) / W3C Semantic Web. Resource Description Framework (RDF). [Online] [Accessed on: December 07, 2009.] <http://www.w3.org/RDF/>.
- [10] O'Connor, Martin, et al. Supporting Rule System Interoperability on the Semantic Web with SWRL. ISWC 2005, LNCS 3729. 2005.
- [11] Gómez-Pérez, Assunción, Fernández-Lopez, Mariano e Corcho, Oscar. **Ontological Engineering**. 2. s.l. : Springer, 2004.
- [12] A transition approach to portable ontology specification. Gruber, TR. 2, 1993, Knowledge Acquisition, Vol. 5, pp. 199-220.
- [13] Gašević, Dragan, Djurić, Dragan e Devedžić, Vladan. **Model Driven Architecture and Ontology Development**. s.l. : Springer, 2006.
- [14] Fikes, R. Multi-use Ontologies. Stanford University. [Online] 1998. [Accessed on: October 03, 2005.] <http://www.ksl.stanford.edu/people/fikes/cd222/1998/Ontologies/tsld001.htm>.
- [15] Guizzardi, Giancarlo. **Ontological Foundations for Structural Conceptual Models**. Enschede : Telematica Instituut Fundamental Research Series, 2005.
- [16] Guizzardi, Giancarlo e Wagner, Gerd. Towards Ontological Foundations for Agent Modelling Concepts Using the Unified Foundational Ontology (UFO). Agent-Oriented Information Systems (AOIS), selected revised papers of the Sixth International Bi-Conference Workshop on Agent-Oriented Information Systems. Lecture Notes on Artificial Intelligence (LNAI) 3508, 2005.
- [17] Guizzardi, G.; Lopes, M.; Baião, F.; Falbo, R. On the importance of truly ontological representation languages, International Journal of Information Systems Modelling and Design (IJISMD), IGI-Global, 2010.
- [18] Research, Stanford Center for Biomedical Informatics. The Protégé Ontology Editor and Knowledge Acquisition System. Protégé. [Online] 2009. [Accessed on: February 27, 2009.] <http://protege.stanford.edu/>.
- [19] Sirin, Evren, et al. Pellet: A practical OWL-DL reasoner. Elsevier B.V. 2007.
- [20] Motik, Boris, Sattler, Ulrike e Studer, Rudi. Query Answering for OWL-DL with Rules. 2005.

Authors

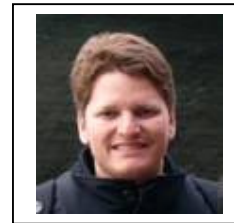
Maxwell E. Monteiro – Doctoral candidate and a MSc. in Information Technology at Federal University of Espírito Santo (UFES). He is currently an Associated Professor at Fundação de Assistência e Educação (FAESA) - Brazil. His main interest area is Intelligent Network Management and Operation.



Anilton Salles Garcia – An Associate Professor of the Computer Science Department, Technical Center, Federal University of Espírito Santo (UFES). PhD in Electrical Engineering: Automation at UNICAMP, has a vast enterprise and academic experience in digital communications in the convergence age. His main topics of research are: Broadband Wireless Access Network, Digital City, Mobile Agents, Network Design and Analysis, Digital Inclusion, Network Management.



Giancarlo Guizzardi – Associate Professor of the Computer Science Department, Federal University of Espírito Santo (UFES). He holds PhD in Computer Science from The University of Twente, in The Netherlands. His main line of research is concerned with the application of foundational ontologies in the development of philosophically and cognitively well-justified methodological tools for conceptual modelling in computer science.



Pedro Paulo F. Barcelos – Graduated in Computer Engineering at Federal University of Espírito Santo (UFES), he is currently a Mastering candidate in Electrical Engineering at UFES. His research area is at applied ontologies to Transport Networks, especially Optical Transport Networks.

