
Project: An Enterprise Architecture Solution for Infrastructure Management**Network Architecture Solution****Pedro Maria Teixeira Dias de Sousa Monteiro**

ABSTRACT

The use of enterprise architecture to describe the structure of how the business processes are supported is already taken into consideration by many companies. Nowadays, it is already possible to obtain different examples of companies that apply frameworks to demonstrate the architecture in an application level to support the business processes. In this research, it is intended to demonstrate the practice of enterprise architecture based on the application of a framework but in the network level. The goal is to focus on the representation of the architecture that supports the application level. Thus, the network infrastructure components have to be well detailed, structured, and modeled in *archimate* – an enterprise architecture modeling language that supports the description and analysis of the existing architecture. In this respect, its support relationship with the application level needs to be well cleared, which in turn is supporting the company's business processes. Nevertheless, it is then possible to complement the network infrastructure component with the existing enterprise architecture, making this component indirectly related to the organizational strategy, objectives, and operations of the company, as well as the practice of enterprise architecture. Based on this methodology (enterprise architecture) and applying an appropriate modeling language, it is also intended to solve specific problems in the network infrastructure component, such as the lack of uniformity in the equipments and the lack of knowledge of the existing infrastructure. Furthermore, the problem will be solved through a centralized, simplified, and holistic solution view of the enterprise network infrastructure.

Author Keywords

Infrastructure, Networks, Views, Equipment, VLANs, Atlas.

INTRODUCTION**Context**

With this document, it is intended to develop an enterprise architecture solution for infrastructure management. However, this research is focused essentially on an area of infrastructure, networks and telecommunications, with the main objective in the development network maps capable of representing in a simplified way the company's network infrastructure.

With these maps it will be possible to acquire a view of where the equipment is physically housed and the respective servers and tools that take advantage of this infrastructure as well as how all these devices communicate with each other, being in different network environments. If there is not a continuous focus on creating and updating the maps of the existing systems in the company, it may cause a lot of management and handling problems in the existing equipment park, reinforcing the relevance of the maps worked in this thesis.

Problem

The infrastructure areas are often focused on solving daily problems unnecessarily when its time could be much better spent in implementation of new projects in that area. This is the main problem that this thesis aims to solve: time lost in solving problems, based on lengthy troubleshooting procedures. The main reasons why, in network infrastructures, companies waste a considerable time on performing repetitive troubleshooting are:

1. The lack of uniformity in network equipments, when it comes to large parks, therefore due to the lack of knowledge of its network structure;
2. Wrong planning for new implementations, as there is no holistic view of all areas where a given change has an impact;
3. Wrong decisions on system updates, or new implementations, coming from departments that do not have such an in-depth knowledge of networks.

To avoid the occurrence of these situations, the implementation of a framework that takes advantage of an enterprise architecture methodology to represent the existing network architecture, is a clear added value for the daily activities of IT (Information Technology) teams. However, the major reason why they continue to deal with these consequences is associated either with the lack of time to understand that they have this need of representations or, even when it is understood, there is a large effort and dedication associated to make it and the team might choose not to dedicate necessarily to continue the update of these representations.

To solve the problems presented, the solution will focus on 3 types of profiles: network technicians, application technicians and strategic stakeholders. Considering the problems described above, the improvement needed for each type of profile were identified, as shown in figure 1 below.

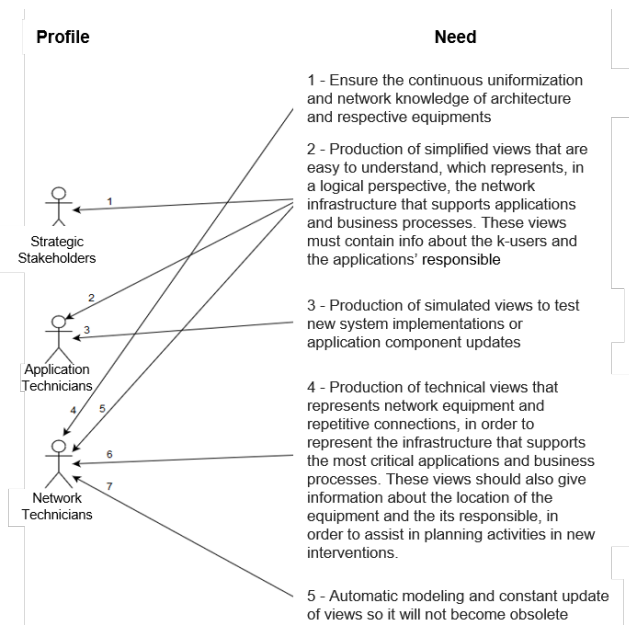


Figure 1. Profile/Need relationship resulting from the problems analyzed, with the identification of the 5 needs and 7 relationships.

RELATED WORK

In order to improve the managerial quality of the infrastructure park, many enterprises apply monitoring tools that give indications of a variety of data on their assets, such as: activity status, processing, storage, temperature, amongst others. These tools provide an inventory of the existing park and a notification feature to proceed whenever there is an occurrence for an asset that is being monitored.

One other possibility is monitoring at the service or access levels or links of connection between the equipment. Monitoring at this higher level can make validations much more complex in large validations parks. For this same reason, it is also important to understand and define what really makes sense to monitor in infrastructure. In the analysis of occurrences during monitoring, it makes sense to take advantage of network maps at this level, since it may be easier to visually understand the source of a given occurrence. Many of these monitoring tools also have the ability to represent the network being monitored, using network maps.

A network map is a diagrammatic view that shows all the equipment and the connections between them (the ones intended to be visualized in a given park), which equipment can be computers, servers and/or network assets that make these connections possible. These maps can detail the location of all these network assets, such as routers, firewalls, switches, VLANs (Virtual Local Area Network), etc., representing the existing connections, and in this way it is possible to create a schematic guide of the entire company network. To prevent the manual work of modeling

a network diagram, some tools with network mapping capabilities have been evolved so that they can take advantage of the Autodiscovery feature, this consists of detecting, in a given network, all existing equipment, connections and access. The feature takes advantage of port-scan capacity to detect all services, which all of these devices are providing to the network. When this functionality is applied, the modeling tool should be able to create small diagram sketches of what detected from the network.

For futuristic effects in the continuity of the network management, after having a well designed and defined map, it is possible to obtain not only a graphical view of what the architecture is and what exists in the network, but also, for troubleshooting purposes, a view from the highest level to the lowest level, which is possible to evaluate a given occurrence from a service delivered on the network, even at the level of a physical interface of a given equipment - used to make a physical connection to another one. In addition, these monitoring tools allow you to make validations to the equipment, in order to check if they already have the latest upgrade developed by the equipment manufacturer and suggest an update if the equipment does not have the latest version yet.

At the moment, it is possible to find on the market several tools with the capabilities previously presented. Two of them are SolarWinds Network Topology Mapper [2] and PRTG Networks Mapping (Paessler Router Traffic Graphic) [3]. These tools are highly endowed and flexible, not only focusing on monitoring and mapping networks, but also at the application level, databases and storage. They allow the creation of different types of maps, where it is possible to define what it is intended to show.

The maps in PRTG Network Mapping contain a set of features, services and hardware that support the applications on a map in a circular shape, from the outside to the inside of the circle, and can provide real-time indications of the status of each of these applications and equipment, such as represented in figure 2 below.

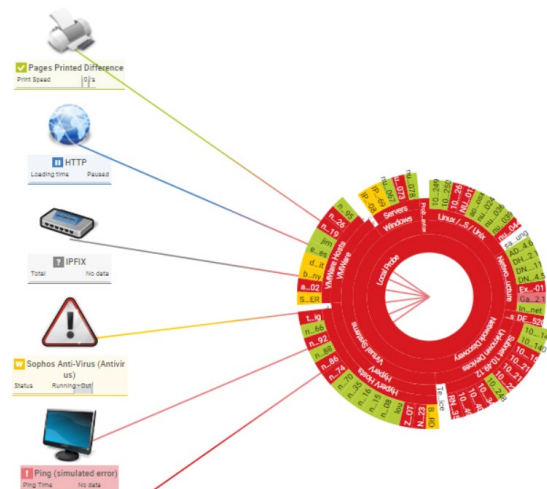


Figure 2. Monitoring view of the PRTG Network Monitor, showing all the services and hardware that support the applications under analysis [3].

Another example of the utility of this tool is the flexibility to create new maps using the previously defined components, as shown in figure 3 below.

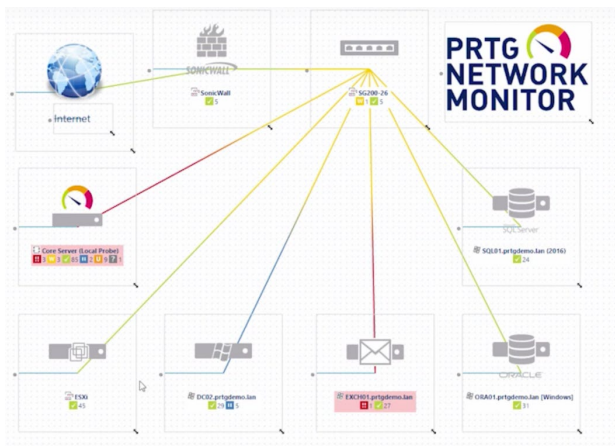


Figure 3. View of the PRTG Network Monitor, with application components and services that take advantage of this architecture [3].

Figure 3 represents how the tools and database servers are networked and how they can communicate with each other. SolarWinds Network Topology Mapper [2], on the other hand, is a tool for mapping and monitoring network equipment where the self-modeling functionality - modeling process done automatically by the tools, through some inputs or minimal data - can go down to the physical level, mapping the physical interface port used by the equipments to communicate with each other. This tool has also the functionality of saving existing configurations on the equipment and apply new configurations remotely.

Figure 4 shows an example of a network map modeled automatically using the Autodiscovery functionality.

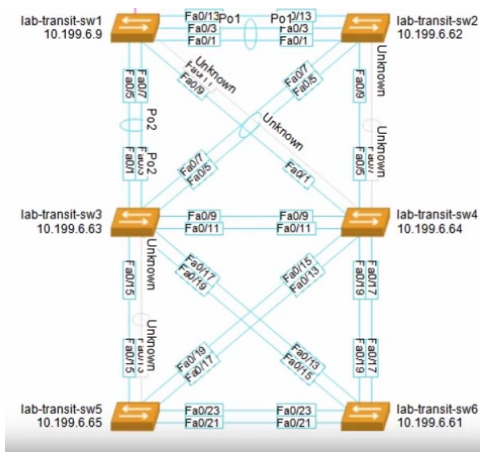


Figure 4. Physical network diagram, drawn using SolarWinds Network Topology with Autodiscovery [2].

In both tools presented, it is also possible to allow the discovery service to be constantly active, or even scheduled, so that any changes are detected and later modelled automatically, thus keeping the models continuously up to date.

For network infrastructure management, SolarWinds Network Topology Mapper is a very useful solution, with a

holistic view of the managed park, working as a clear guide for troubleshooting and monitoring, as well helping teams in uniformization and in configuration management of the existing equipments.

In fact, having the Autodiscovery always working to ensure that the map is always up to date would make perfect sense. However, to ensure all equipment and services are detected, many probes are needed across the network and all of this needs to be allowed to make constant scans within the network. These scans can congest the network and often, the services used to perform it, could be interrupted due to the company's security policies applied in firewalls.

The problems presented previously - the lack of uniformity, or the lack of knowledge of the existing equipment park presented by the areas of network infrastructure - can be easily solved applying SolarWinds. All equipment can be inventoried in the tool and the existing configurations can be found in each of them, thus making it easier to maintain the uniformity of the park, as represented in figures 5 and 6.

Name	Polling IP Address	Status
bas-2821-2.aus.lab	10.199.4.3	Node status is Up.
bas-2851-local	10.199.4.4	Node status is Up. Hardware health monitoring is in Critical state.
bas-2911-vg02	10.199.4.17	Node status is Down. Hardware health monitoring is in Undefined state.
bas-2948G	10.199.4.13	Node status is Up.
bgp-2651-03.aus.lab	10.199.3.6	Node status is Up. FastEthernet0/1 - link to bgp-2651-02 test 1 ok ok 1231
cur-2621-aus.lab	10.199.3.4	Node status is Up.
cur-2851.lab.cur	10.199.3.5	Node status is Down. Hardware health monitoring is in Undefined state.
cur-3550	10.199.3.10	Node status is Up.
cur-3750	10.199.3.11	Node status is Up.

Figure 5. Equipment inventoried and monitored by SolarWinds Network Topology [2].

NODE NAME	DATE TIME	View Change Report
bas-2911-vg02	9/28/2016 2:01:29 AM	View Change Report
bas-2911-vg02	9/28/2016 2:01:10 AM	View Change Report
bas-2911-vg02	9/27/2016 2:01:41 AM	View Change Report
bas-2911-vg02	9/27/2016 2:01:25 AM	View Change Report
bas-2911-vg02	9/26/2016 2:01:30 AM	View Change Report

Figure 6. Previous configurations, of a given equipment monitored by SolarWinds Network Topology [2].

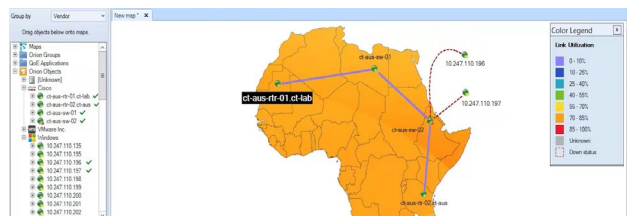


Figure 7. Monitoring of equipment and respective links between them, through SolarWinds Network Topology [2].

For the purposes of new implementations, it is not always easy to understand the impact associated with a given change, even when applying network maps. Figure 7 represents a view with a set of equipment and the respective links between them, which include routers, switches and servers. In case its intended to make a direct or indirect change to one of the equipment shown, it is good to have in mind that if this change is made at a certain time of the day, it can cause a lot of inconvenience to the business operation that is using a certain service. This view can be very useful, however, it was not possible to obtain a clear view of its use

at the level of business or application processes levels - it is still missing information regarding business areas associated with the use of a given service, or even demonstrate the level of dependence that some business applications have on a given service, in other words: whether they can work offline or not.

On the other hand, as mentioned before and as we can see in the figures presented in this chapter, these views are not easy to understand; it is necessary to have a notion of the nomenclatures used, manufacturers and functional utilities of the equipment or services. It is complicated to generate with these tools views that allow strategic areas or applications to simulate new projects where they can easily understand what they are watching, and if the requirements of their projects are easily reflected in these maps. As indicated in the profile/need relationship 2 in figure 1 of chapter 1, in order to plan the implementation of a new application, it is relevant to understand how, and, whether a given protocol can be used in the company's network - due to application requirements, and the answer might be as simple as: may or may not - in this case whoever is doing the planning does not need to understand if it is the firewall, for example, that is allowing them to use this protocol.

A simple methodology, easy to understand and useful for representing IT systems, to bridge the association with business processes, is enterprise architecture through the archimate modeling language [4]. This methodology uses frameworks to represent views that align the company's strategy, processes and information with the IT assets that support the business. With the archimate language, these views can be modelled from a higher level, where it could represent layers that support the business process - with the application layer and finally the infrastructure layer, as shown in figure 8.

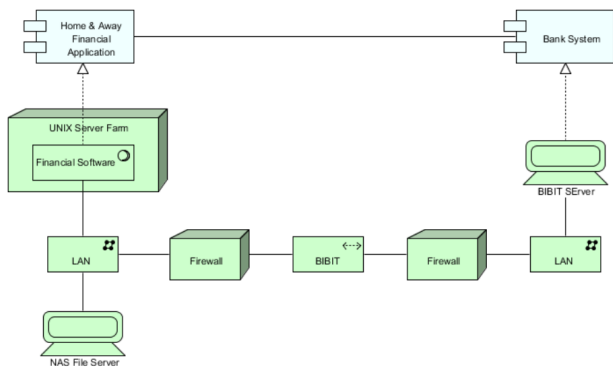


Figure 8. Archimate view with the infrastructure that supports the application components shown in the image [5].

With this type of modeling it is possible to make views of different styles, for example, in the case of infrastructure, to represent the map in a logical aspect - depending on the configurations of equipment associating it with the respective services; or physical - representing the real connection made between the equipment. An example of a modeling tool that is based on frameworks and enterprise architecture is Archi [4]. It was through this tool that the proposals for views presented in chapter 3 of this document were developed. Tests were carried out with the tool, in order to validate whether – based on the components of the

archimate to be represented – it would be able to generate the ideal frameworks that would solve the needs presented in figure 1. Archi is a modeling tool containing no type of capacity of Autodiscovery and monitoring, or even integration with other tools - its functionality focuses only on the design of models in archimate.

Another example of a tool that uses archimate, as an easy-to-understand methodology for modeling maps, is Atlas [6]. This tool has a strong component of adaptation to different realities. For the representation of a given equipment or service for future modeling, the tool shows some freedom for the definition of new classes (it works based on classes to represent the type of equipment or services), subclasses and dependencies that aim to associate between these elements – even if they are not available in the basic tool manufacturing options. In addition to this component, the tool is capable of some of the aforementioned capacities, however it needs a good preparation in the first phase, due to its great adaptability. It also implies some initial effort to configure these analysis capacities in the equipment, and in the production of representative maps, which will be explained in more detail in the proposal chapter.

Table 1 represents a list of capabilities that would be relevant to obtain under this research through the tools presented (SolarWinds, PRTG Network, Archi and Atlas).

Capabilities
Monitoring
Network Maps
Freedom to adapt and configure maps
Autodiscovery
Validation of the need to upgrade assets
Asset configuration history
Semi-auto-modeling (no network scans)
Business Architecture Methodology
Archimate language for modeling maps

Table 1. List of relevant capabilities

Considering the 5 needs presented in Figure 1 of Chapter 1, it is possible to see in Table 2 below which tools are capable of solving the needs of each profile.

Figure 1 needs'	Solar Winds	PRTG Network	Archi	Atlas
1			X	X
2				X
3	X	X		X
4	X	X		X
5	?	?		X

Table 2. Comparison of need resolution capabilities between the previous tools.

It is part of the problem-solving process to explore infrastructure management in the context of enterprise architecture, so Altas would be the natural solution. This

tool was chosen for the development of this project and the reason for this decision is not only because of the references in table 2, but also due to the fact that the tool was previously selected for the development of this project. For the case which it is intended to apply the solutions that will respond to the 7 profile/need relationships presented in Figure 1 of Chapter 1, it will make sense to use Altas, because, in addition of being a tool that makes it possible to explore several domains, there was neither information nor sufficient experience regarding other tools. However, as seen in the need number 5 in table 2, it was not possible to confirm if there were other means of view update.

In order to make it clear how the Atlas approach all these capabilities, it will be discussed more detailed in Chapter 3, in the proposal of this document, indicating solutions or ways of working as work-arounds for Autodiscovery cases, self-modeling and ways of updating views.

PROPOSAL

The enterprise architecture modeling language used in this thesis is archimate. The 3 types of stakeholders identified previously are network technicians, application technicians and stakeholders from a more strategic area. Therefore, the styles of views considered for the different profiles must have some common interconnection base. Consequently, in the first place, it is intended to define a simple modeling language, to demonstrate to the application and strategic areas how its processes are supported at the level of network infrastructure; and second, to define a common base as a mean of linking between views dedicated to network technicians and those dedicated to application technicians. Since this project focuses on problems associated with the network or dependent of it, the research will consist on trying to define a common base (for any and all areas), which will be the VLAN.

All large and well-structured network parks have their internal networks segmented into shorter network segments, each of which has its own requirements, objectives and areas. It is intended to consider the concept of VLAN as an environment class in archimate, for any and all equipment or applications. Each VLAN has its requirements in terms of: network configurations [IP (Internet Protocol), netmask and gateway], connections allowed to other networks, traffic allowed on these connections and permitted network protocols (where each equipment associated to a given VLAN, will assume its requirements at the network level).

Ensuring the uniformization of the network equipment and the respective architecture view

It is relevant to represent, at a more technical level, how these VLANs are published on the network, how equipments and applications are associated with it, as for represented in figure 9, which will be explained later.

In the case of network technicians, as indicated in the profile/need relationships 4 and 6 of figure 1 of chapter 1, in order to solve: the problem associated with the lack of uniformity of the network parks (for knowledge of the network architecture) and the need for support in planning interventions; its proposed a view to show them the physical location of the equipment, the most critical business

processes associated with these networks and those responsible for them. Figure 9 below represents an example.

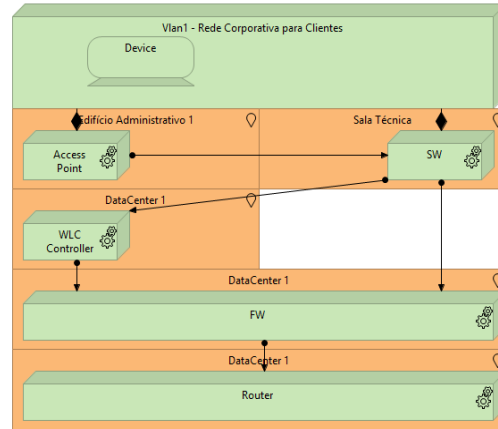


Figure 9. View proposal developed in archimate, to represent the network equipment that support Vlan 1.

Vlan1 - Rede Corporativa para Clientes (Node)		
Main	Name	Value
Properties	ID	1
Analysis	Nome	Rede Corporativa para Clientes
Appearance	Range IP	192.168.0.1 - 192.168.0.254
Figure	Gateway	192.168.0.1
	Mascara de Rede	255.255.255.0
	Responsável	Membros da equipa de redes

Figure 10. Lists of properties and respective values of Vlan 1.

Figure 9 demonstrates the network architecture that supports a given VLAN and shows how it is published on the network.

For each of the network elements represented in figure 9, there is a list of properties associated with each of them, as in the example of Vlan 1 in figure 10. These properties are requirements in the creation of the element provided by its class, with the purpose that each of them has its own class, with its respective properties.

Figure 9 represents a simplified form of an architecture. However, the objective is to demonstrate how the constitution of a VLAN works, its dependencies in terms of network equipment and the physical location of the equipment to facilitate the resolution of errors and planning of necessary interventions. Through the style of views in the figure, it will then be possible to guarantee the resolution of need 1 and 4 indicated in table 2 of chapter 2.

Development of simplified views with logical representation of the infrastructure

In order to solve problems associated with difficulties in understanding the network structure on the part of the application areas, efforts was made to design a simple way to explain how VLANs can communicate with each other, trying to represent the rules associated with these communications. As a proposal for this modeling it is intended to use the Path class of the archimate to represent this communication between VLANs and the respective connection rules, as shown in figure 11.

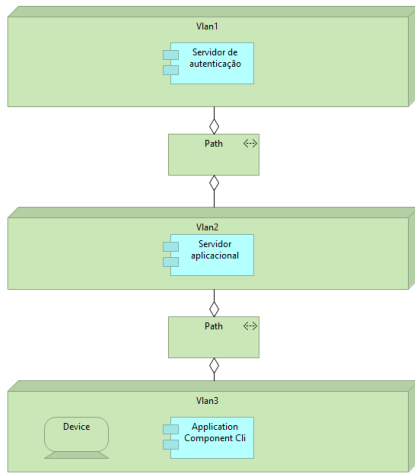


Figure 11. View proposal developed in archimate, to represent in logical form the existing network infrastructure that supports a given application and its components.

↔ Path (Path)		
Main	Name	Value
Properties	Vlans	Vlan1-Vlan2
Analysis	Portos	22-8080 22-443
Appearance	Protocolos	tcp udp
	Trafego	15 MB/s

Figure 12. Lists properties and respective values of the Path class represented as an object in figure 11.

Assuming the concept of VLAN as an environment where applications or equipment can be associated, as previously described, in this view there is an application system in different environments, choosing the most appropriate VLAN for each component regarding its objective and how the components can communicate, being in different network segments.

The obstacle is to find a way to represent the characteristics of these VLANs and the communication rules existed between them. Therefore, the idea is to try to represent these rules between VLANs as a communicating vessel of connection between networks. This vessels are used to indicate if a given device can or can not send data to an application component, being associated to different VLANs. On the other hand, in the absence of this communicating vessel between two VLANs, the equipments associated to different VLANs can not send or receive data. All equipment and components that are associated with the same VLAN can communicate with each other, without any restriction, without needing a connection representation (communicating vessel) between them.

The Path class represented in figure 11 demonstrates the possible communications between VLANs, symbolizing the communicating vessel concept described before. However, this communication has restrictions imposed by the Path of the connection through firewall rules, for example.

The referred restrictions can be seen in the properties of this Path, represented in figure 12. Therefore, the representation of these restrictions will be possible make, being necessary to expand its Paths in different views to understand the restrictions imposed, as represented in figure 13.

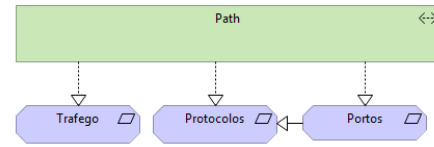


Figure 13. View proposal developed in archimate, to represent in logical form the restrictions of a given Path between two VLAN's.

↔ Protocolos (Requirement)		
Main	Name	Value
Properties	TCP	22-8080
Analysis	UDP	22-443
Appearance		

Figure 14. Lists of properties and respective values of the Protocol class represented as an object in figure 13.

As in the application scope, most of the requirements of applications at the network level are traffic, protocols and communication ports, it makes sense to organize and group them by classes, thus facilitating the understanding of the rules of the connection.

Each of these restrictions has an associated list of rules, referring to all types of communications at the level of protocols, ports and traffic allowed between equipment, or application components that are on VLANs that share that Path. With this style of view, it becomes, therefore, easier to understand how the existing network infrastructure works logically and how it is possible to take better advantage of it without making involuntary mistakes due to wrong decisions.

On the other hand, this way of associating applications with VLANs allows network technicians to be informed about the applications that depend on the VLANs they are managing, as indicated in the profile/need relationship 5 in figure 1 of chapter 1. In addition, all applications will have, as a property, a reference to the responsible or k-user of the same. Any intervention by network technicians that creates an impact on any VLAN can be better managed, as it will be possible to obtain information regarding the applications associated with those VLANs, as well as the respective responsible. From a planning perspective for these interventions, it will be very useful to know these k-users in order to understand the business processes that are supported by the applications, making it possible to better define the time period for a given intervention. Through the style of views presented in the example of figures 11 and 13, it will, therefore, be possible to guarantee the resolution of need 2 indicated in table 2 of chapter 2.

Automatic modulation and view update

A relevant point - associated with the fact that Atlas does not have Autodiscovery features - is to look for a way to update existing maps automatically, to avoid the effort associated with this update (the main reason why all the problems described in this research still persist in companies) without relying on probes that scans the network to detect equipment, as explained above. In order to avoid constraints on the network, or that these scans (caused by Autodiscovery) are not blocked due to political restrictions

of the existing firewalls, and trying to meet this cause (effort associated with updating the maps) and as indicated in the profile/need relationship 7 of figure 1 of chapter 1, it is proposed to create forms that allow the user to indicate the corrections or the equipment that they intend to add, showing the dependencies to other equipment that may need to be corrected as well. After completing this form, the modeling tool should simulate a network map depending on these changes. If this simulated map goes accordingly with the needed changes, it has to be approved and the network technician must be notified to apply the settings proposed in the form.

At this point, we would be able to not only ensure that our views do not become obsolete, but also a way to save all corrections or implementations of new projects, due to filling out these forms. With these forms it should also be possible to define the date when the proposed changes will be applied and whenever a given proposed change is approved and the same applies in the present time. The tool must keep the views and the status of the equipment before and after changes are applied, that way, the change historic will not be lost, getting a sense of how the infrastructure was in the past, a map of as-IS and one of to-BE.

As indicated in the profile/need relationship 3 in figure 1 of Chapter 1, the development of simulated views, in order to test new system implementations or application component updates, is a necessity and will be possible with this update functionality, through the registration of a new projects. By implementing the forms described in this chapter and using the planning functionality to obtain the to-BE maps, it will then be possible to guarantee the resolution of needs 3 and 5 indicated in table 2 of chapter 2.

Limitations

The characteristic of freedom of adaptation presented in table 1 of the previous chapter, allows the development of other capabilities as extras (which were also referenced in table 1), such as: monitoring, validating the need to upgrade equipments and the history of network asset settings. Atlas has a small disadvantage to develop these capabilities: the initial configuration effort associated to develop it is major. All of these capabilities are possible to implement in Atlas, however, they do not come natively as functionalities of the tool, having to be prepared and configured in advance. Despite this, it is a tool that can be easily integrated with other tools - if the effort associated with all this configuration does not pay off, Atlas can be used only for management of the network maps. Features, such as asset monitoring management and asset configuration management can be delivered to specialized tools. The integration with these specialized tools will allow to Atlas use the information that is not on it management domain.

In the practical case of this research, the proposed solution will be applied to a company that already contains tools suitable for carrying out the functionalities mentioned in the previous paragraph.

The company contains a monitoring tool for all network equipment and connections between them, and yet another tool that manages the settings applied to the equipment, with their history, also validating the need to upgrade each one. Since all of these tools are specialized for the specified

features, and correspond to the needs of the company for the same purpose, the decision was made to hand over the management of the respective features to the tool, since this way would never compensate for the development effort in Atlas.

Another constraint already mentioned in chapter 2, and in the previous subsection, is the fact that Atlas does not contain Autodiscovery functionality. In order to overcome this limitation, it is intended, only in an initial phase, to complement this project with a tool capable of scanning the network, which it will be used only once to carry out the initial identification of network assets.

WORK IMPLEMENTATION

As mentioned in chapter 2 of this research, the tool chosen to work with the solutions presented in the proposal in chapter 3 was the Atlas. All figures and tables that will be presented in this chapter will help to demonstrate the implementation of the proposed solutions, taken from Atlas, which all the data used to produce the completed results were used with the same tool.

Atlas classes' preparation

The views in Atlas are based on previously created blueprints, which are generated through simple algorithms, which process the results through relationships between the objects and the respective data, depending on what is intended to be obtained with the blueprint. These objects must be associated to a given class with well declared fields. To generate the modeling through blueprints, the pre-loaded classes in Atlas were not prepared for the modeling needs of this research, as it is intended to represent a very specific environment within the infrastructure. Some fields of the existing tool classes were edited, and new ones were added, such as the example in figure 15, to allow the representation of all types of objects presented before in the proposal.

Class - Switch

Na...	Lab...	Ba...	Restrictions
Firewall	Firewall	changed by	Firewall
Gateway	Gateway	Text	
IP	IP	Text	
Ligação	Ligação	flows to	Switch
Localizaç...	Localizaç...	located at	Localização

Figure 15. Switch class with its properties and relations to other classes. The relationships are specified in the “Restrictions” field.

The relations represented as “restrictions” in figure 15 are necessary to make it easier for the blueprint algorithm to navigate from class to class through these relations to identify the desired objects during the blueprint processing.

Autodiscovery of the network equipment

After defining which classes to be used, it will be necessary to prepare the data with the equipment and respective

characteristics. In the practical case of this research, it was not necessary to use an autodiscovery tool, since the inventory of equipment, connections and configurations already existed. It was only necessary to organize and centralize the data. Regarding the scope of application and network requirements, it was also not necessary to use the tool, since only a few test examples were chosen, which application requirements were already known in the company and some data associated with the test samples were taken directly from the productive company firewall rules.

Prepare the discovered data and import to Atlas

After selecting all data sources, retaining and centralizing them, it is necessary to prepare its import. All the data were prepared and worked in Excel in the format required by Atlas, so, when importing the file, there is a direct association of the file fields with the properties defined in the respective classes. During the import process, objects are automatically created and stored in the tab of their respective class.

Implementation of the planned views in Atlas

The Atlas contains an editing section to create and generate blueprints, which is possible to define the view to be generated. Through this editor, it is possible to define variables, apply property filters to it through queries and show other types of classes through the relationships with these variables in order to show the results with the blueprint.

Representation of the physical network device connection:

To generate a blueprint for network technicians, it was decided to show all the path where the network flow goes from the connection request of a given client application to the router that allows communication be delivered outside the existing internal network. This implemented blueprint contains the client application as a variable and it is intended to represent the structure that supports the VLAN used by this application as represented in figures 16 and 17.

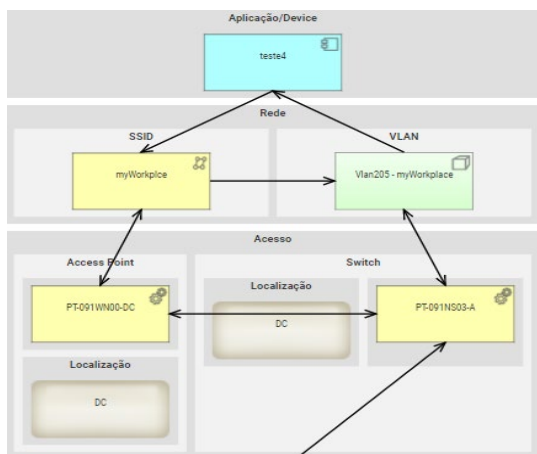


Figure 16. Result of the blueprint with the network structure that supports the connection of the test4 application (Part1)

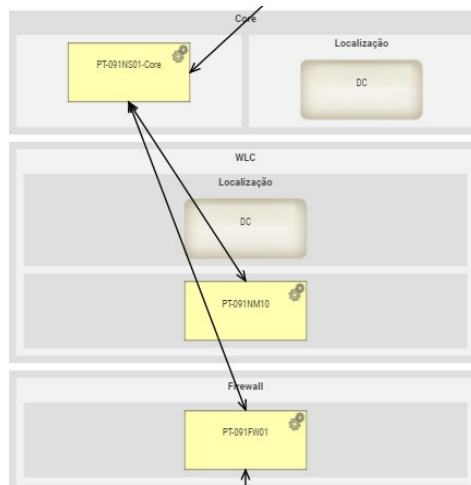


Figure 17. Result of the blueprint with the network structure that supports the connection of the test4 application (Part2).

It is important to note that the connection status of the application (associated or not associated), the IP of the same, the SSID (Service Set Identifier) to which the application is associated or trying to associate, and the AP (Access Point) used to perform this association, are fully automatic data and it's achieve directly from the WLC (Wireless Local Area Network Controller), the equipment that manages APs, represented in figures 16 and 17. The referenced data are updated in the Atlas objects every 5 minutes based on a script that performs queries on the WLC, prepares the data to Atlas requested format, and import it into to the Atlas repository.

Floor plans with the Access Point physical location:

In the case of figure 16, presented in the previous section, some applications/equipment are associated and authenticate on the network via Wi-Fi through an AP. Some of the difficulties encountered during the resolution of problems in warehouses includes the identification of the physical location of the APs, especially for those who are providing remote support and do not know the existing building plans and logistic areas where the equipment is installed. To prevent this situation, were created views to represent the warehouse floor plan with a clear representation of each of the areas known by the Logistics that manages the building, as well as a view with the clients associated with each AP, as in the example in figure 18.

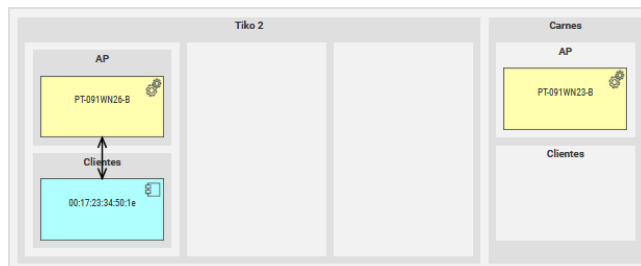


Figure 18. Extraction of the warehouse plan with representation of the existing areas, the APs, and also applications associated in area Tiko 2 (freezer chest of the warehouse divided by 3 aisles) and Meat (meat chest of the warehouse with a single aisle).

With the implementation of the maps in figure 18, in addition of helping the remote support teams, it can also be useful for the logistics area, facilitating the operation analysis.

Simplified logical representation of the network infrastructure:

The blueprint in figure 19 has two application variables, and it is intended to demonstrate whether there is actually a connection between the two applications, and if the existing network allows the applications to share data between them, taking into consideration the application requirements for each of them and the path that may exist between different networks for the data flow.

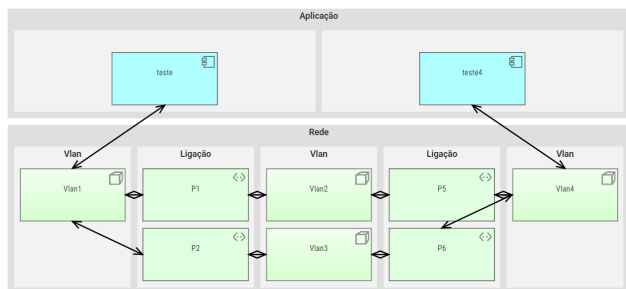


Figure 19. Result of the blueprint used to represent the logical connection between application teste and teste4.

This blueprint shows two possible flows to represent the data that is forwarded between Vlan1 and Vlan4, and as it can be seen, the possible paths may be through Vlan2 or Vlan3, which either both are allowed to communicate to Vlan1 and Vlan4. This means, therefore, that Vlan1 and Vlan4, are isolated between them, but not with the other networks.

As mentioned in chapter 3, applications assume the characteristics of the network to which they are associated, including the gateway, mask and IP of the existing range. However, it is not the VLANs that specify the routing of traffic, but the Paths. In order to be clear and understandable the characteristics/requirements of the network existing in the connections between VLANs are represented in a second blueprint that was developed to represent the Path. In figure 20, the result of the blueprint Path is shown, for connection P1 in figure 19.

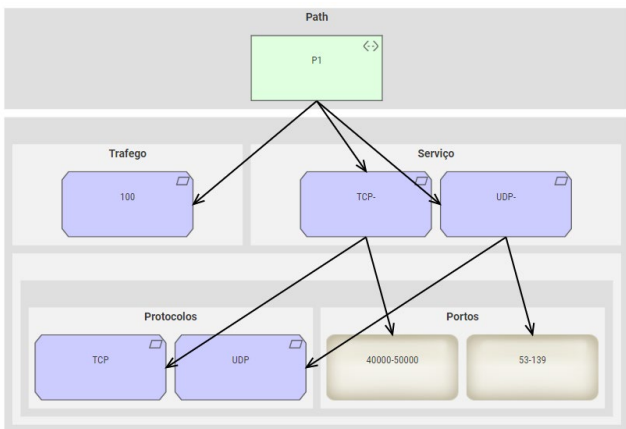


Figure 20. Result of the blueprint used to represent Path P1.

Since Path P1 is the connection between the Vlan1 and Vlan2 networks, as specified in figure 19, the requirements specified in figure 20 are also requirements applicable to traffic between the two networks. As it can be seen in figure 20, it is stated that the P1 connection has a maximum speed limit of 100 Mbps (Megabits Per Second) and only allows the “TCP-” and “UDP-” services, which include only the respective protocols and the range of ports specified.

If the applications to be validated are associated with the same VLAN, the only inherit characteristics are the characteristic of network to which they are associated, with none types of requirements mentioned previously being applied, as there is no connection with a Path to be applied.

Forms implementation and views update:

In the blueprints presented above, the views generated are dynamic, with no general map of the network. The view is generated at the moment and the map is being represented according to the content searched, depending on the respective chosen blueprint variables. The theme treated in this section was interpreted in a different way from the one proposed in chapter 3, because here the modeling of the view starts to be performed at the time is asked. This way of modelling is already automatic and, on the other hand, will not be the views that needs to be updated, but the existing data, which will be used to generate the maps.

To update the data in Atlas, forms are applied and executed, which also serve as guides for technicians to apply the new configurations to the equipment after approval of the proposal of the executed forms. In order for this update method work in a practical and safe way without compromising the smooth functioning of the architecture (since the implicit changes in the execution of forms are soon applied to the respective objects), Atlas has a simulation feature so that it is not necessary to perform tests changes with productive data and maps. The “Scenario Manager”, is a section that allows creating scenarios, importing data from the productive repository and executing the forms in environments apart from the productive. This method allows to any of the profiles identified in the introduction of this thesis, to have the freedom to test any type of changes where the impact on infrastructure could be simulated through the visualization of maps in blueprints.

In Atlas' Scenario Manager, it is possible to create and save a scenario of the infrastructure status in the past, in the present (as-IS) and in the future (to-BE), thus obtaining a time history of the infrastructure.

CONCLUSION

Regarding the identification of needs carried out to solve the problem described in section 1.2 of this research, we are confident that they are a reality for many companies. This can be observed by me due to the fact that, as a student, I also work in the area and many of the situations described in this document are part of my daily life. As an end customer of the solution (like my case), I tested the tool developed in the company where I work, and I am convinced that the proposed views can help considerably in solving the problem and will be an advantage for the future governance management of the IT department. It will certainly be something to prove with a more constant use of the

developed implementation. For the tests conducted in the enterprise, a set of problems were defined, and the support of Atlas was used for the resolution; it was possible to verify that, in a troubleshooting perspective, the functionalities generated in Atlas were an important tool. It was observed that the tool only solves totally, some of the defined problems, since Atlas is being used only for the purpose of validating, visualizing and analyzing data and views, serving only for part of the process of solving a given problem (troubleshooting). In order for the Atlas functionalities to have the full capacity for solving completely all of the problems, it would be necessary to interact with equipment, systems and clients, allowing the configuration commands to be sent to the equipments, preventing that part of the problem resolution be treated directly on the equipment (made by a technician).

The interaction with equipment and systems through Atlas is not a totally impossible feature, but, as explained in chapter 3 of this document, as with other Atlas capabilities, it is not a feature that is natively created in the tool, and as such it would require considerable effort for its development. Bearing in mind that this research also aims to develop simplified views of the infrastructure for application areas, we would be creating a greater risk for possible errors if would be allowed the configuration of more sensitive and complex equipment, which could compromise the company's infrastructure. The configuration of network equipment requires specialized knowledge and, although the functionalities generated in this project facilitate the understanding of the network infrastructure for those who are not specialized, they may not be enough for the front line teams or application areas to know exactly what that they will be dealing with, if they have the possibility to correct or change equipment settings. For that reason, the direct interaction functionality would have to be studied in depth, in order to prevent as few errors as possible, and this was another reason why this functionality was not proposed.

Another relevant feature, also associated with the direct interaction between Atlas and equipment, is the automatic update of existing data. With the exception of objects of class client (which connect to the network via Wi-Fi), AP, SSID and IP (IP addresses used in Wi-Fi networks), all other objects have to be updated based on the forms generated in Atlas. Any changes intended to the network infrastructure will have to be suggested, validated and corrected in the Atlas and finally carried out directly by the network specialists on the equipment (accessing directly to the equipment). This procedure can cause sometimes a delay in the process, but it ensures that the maps do not become obsolete and that they remain valid, which is a strictly necessary process. The information of the equipment connected by Wi-Fi taken from the WLC (controller of APs), are examples that it is possible to automatically update data in Atlas, as indicated in the explanation of figure 19. The script created to export the existing data from APs and WLC to the Atlas is very specific. Therefore, exporting the data from other equipment would require a lot of time, since each equipment has its own language for interaction, not

being possible with the planned period for the development of this thesis.

From a futuristic point of view, considering that the project is not entirely complete, it was able to help with solving the identified problems in the company in which the project was applied. I believe that this research contributed to proving one of the concepts intended, since it represents in detailed and useful way the network infrastructures in archimate. Lining up the created maps to the application layer, which in turn, would already be linked to the process layer, implies in this way the practice of business architecture methodologies in the areas of infrastructure, specifically in networks and telecommunications.

REFERENCES

1. D. Arnott and G. Pervan, "How relevant is fieldwork to DSS design-science research?" *Frontiers in Arti_cial Intelligence and Applications*, vol. 212, no. January, pp. 108-119, 2010.
2. ITPRC. Retrieved 2019/05/28 from <https://www.itprc.com/best-network-diagram-mapping-and-topology-tools/>
3. Link Consulting. Retrieved 2019/05/28 from <http://www.linkconsulting.com/atlas/>
4. Paessler AG. Retrieved 2019/05/28 from https://www.paessler.com/prtg?gclid=Cj0KCQjwla7nBRDxARIsAD1l0kCXOnQzo8GNWwxhY4EQ1DjzL6l2J1vtglgL4OK1mBuS59E1vPLgBOEaAu1_EALw_wcB
5. Phillip Beuvoir. Retrieved 2019/05/28 from <https://www.archimatetool.com/blog/2018/10/05/why-archi/>
6. SolarWinds. Retrieved 2019/05/28 from <https://www.solarwinds.com/pt/>
7. Visual Paradigm. Retrieved 2019/05/28 from https://www.visual-paradigm.com/support/documents/vpuserguide/4455/4456/86484_technologyus.html