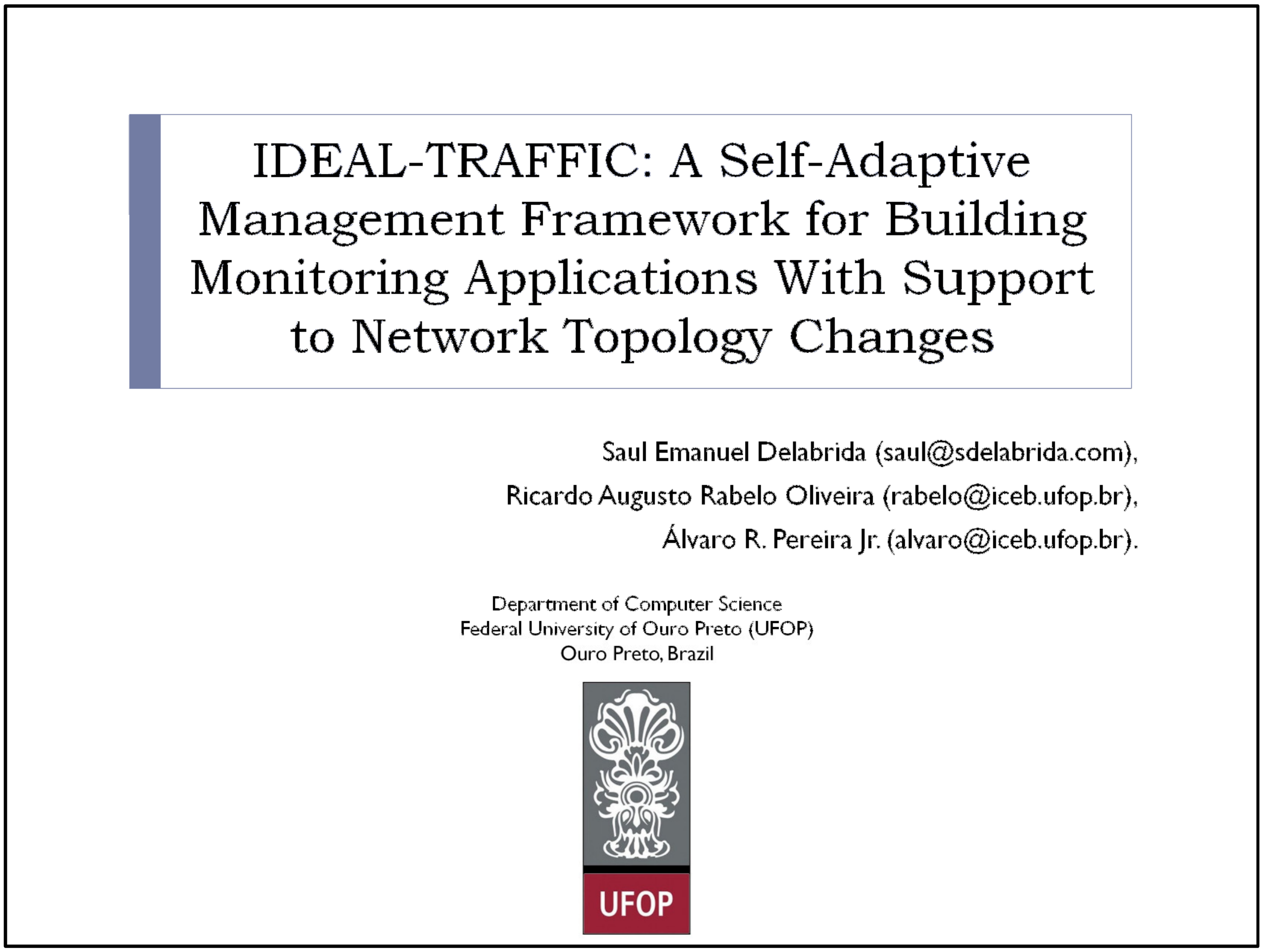
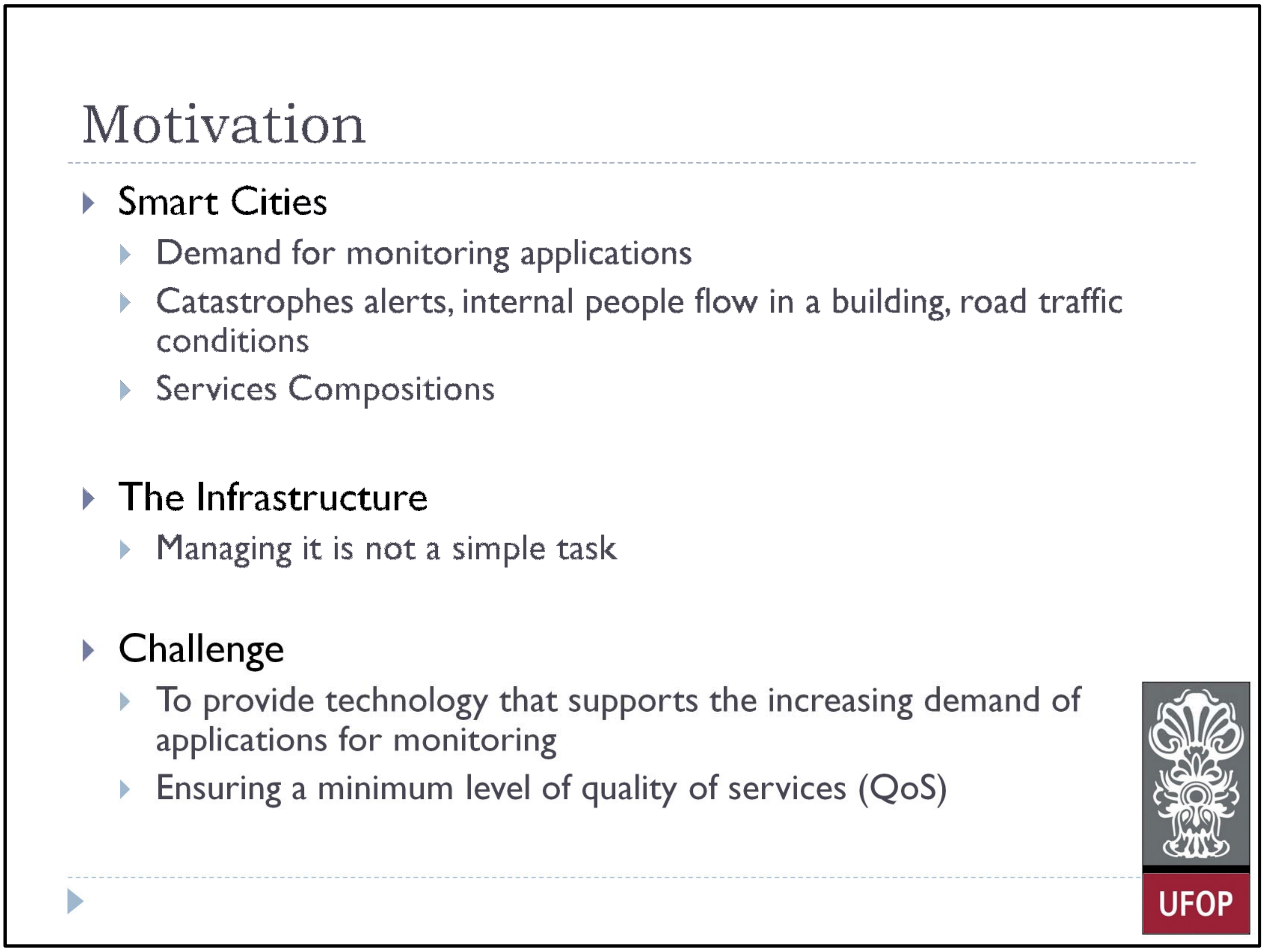


The evolution and dissemination of network communication technology and the advanced status of embedded devices encourage the creation of solutions for monitoring cities in various environments. End-users can benefit from applications that deliver information in real time. On the other hand, administrating these applications is not a trivial task. Components may fail and invalidate an application. In this essay we present IDEAL-TRAFFIC: a framework based on SOA architecture with modules to manage the state of the service. IDEAL-TRAFFIC provides a simple interface that enables the system administrator to create applications and to make them available to the user. A self-adaptation process is necessary in order to ensure quality of services. In order to maximize resources usage, a process of readaptation is embedded. These approaches follow the rules of the application's requirements and do not depend on human intervention.

Outline

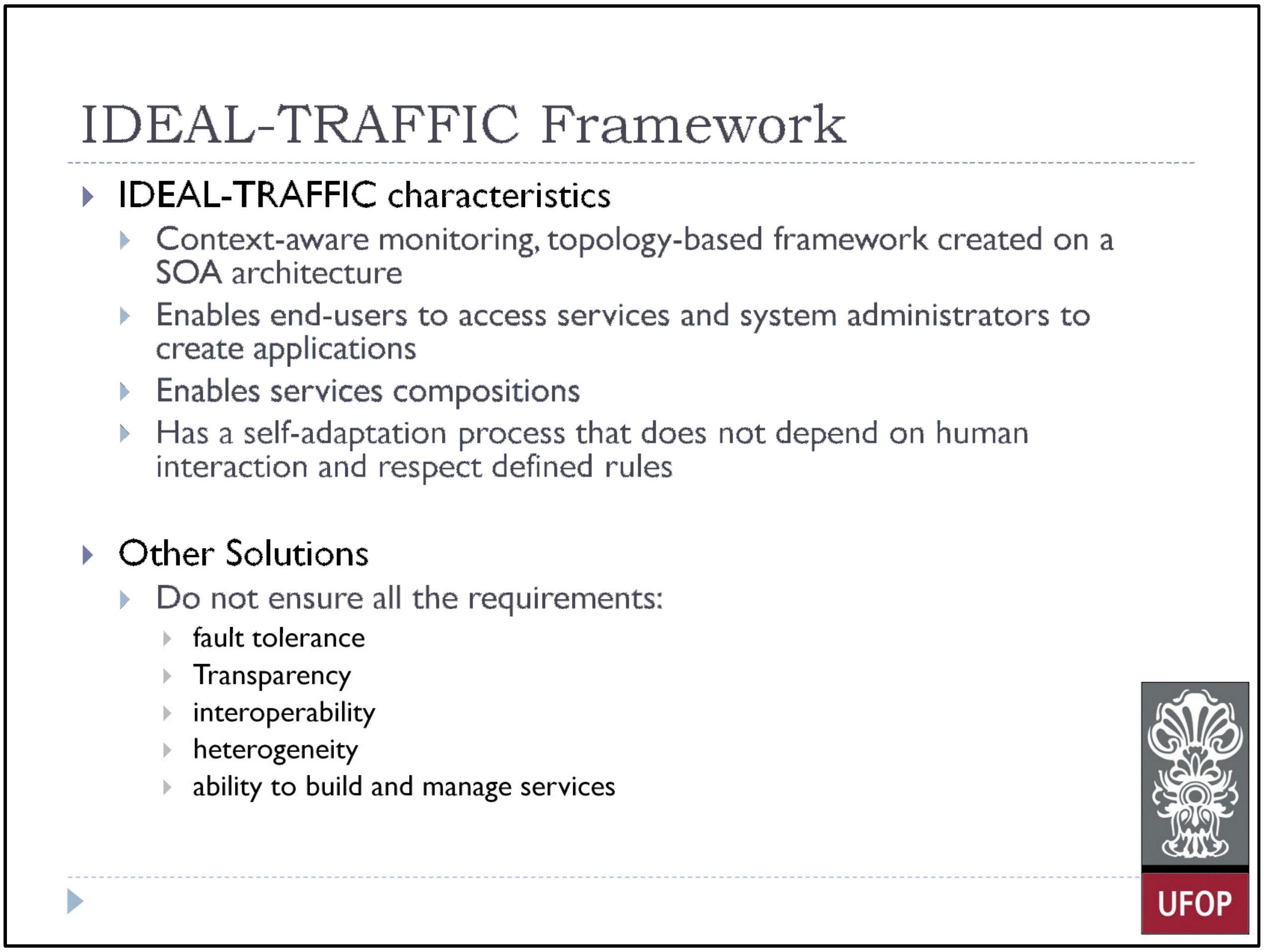
- Introduction
  - Motivation and Project Goals
- IDEAL-TRAFFIC Framework
  - Definitions
  - System Requirements
  - Architecture
- IDEAL-TRAFFIC Management
  - Management Requirements
  - Service Creation Process
  - Adaptation Process
  - Readaptation Process
- Use case
  - Traffic Radars

UFOP



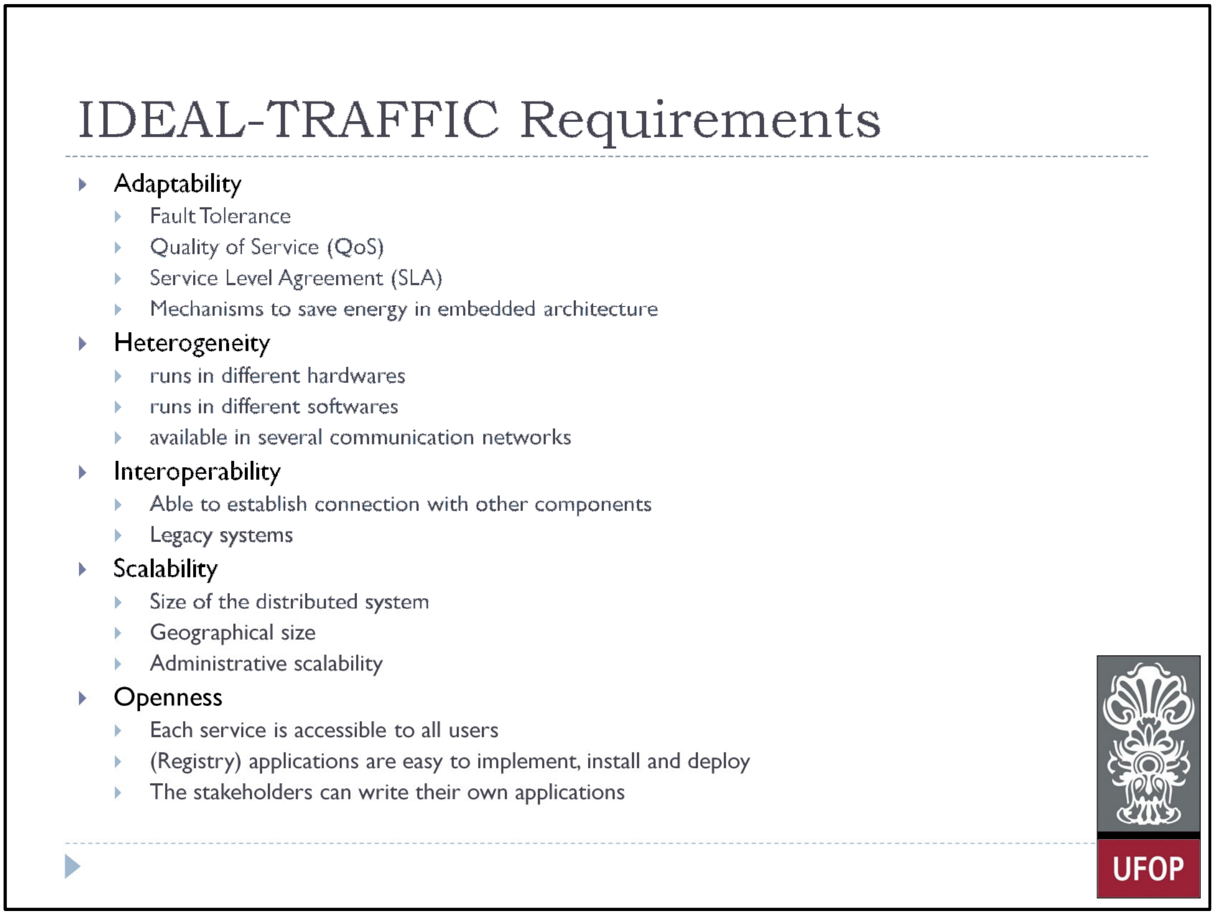
The demand for monitoring applications has increased in recent years. 'Smart cities' is a term used for an emergent area composed by several sub-areas which generate researches and novel solutions for user's convenience. Catastrophes alerts, internal people flow in a building, road traffic conditions, among others, are examples of services that could be provided by monitoring infrastructures.

Managing this infrastructure is not a simple task. The challenge is to provide technology that supports the increasing demand of applications while ensuring a minimum level of quality of services (QoS). Furthermore, solutions found need to provide fault tolerance, transparency, interoperability, heterogeneity and ability to build and manage services.



Therefore, we introduce the IDEAL-TRAFFIC framework, based on SOA architecture. IDEAL-TRAFFIC is a framework that allows users to access available services, and provides system administrators with an interface to build Services Composition (SC) that makes up other applications. IDEAL-TRAFFIC is a topology based network, which enables the building of applications in real time without installing hardware or software, and without turning it off, except at the discretion of the system administrator.

Another important feature is the self-adaptation process which does not depend on human interaction. Although the system administrator can define rules to carry out the adaption process, IDEAL-TRAFFIC has autonomy to perform changes based on these rules. The network topology status is used in the adaptation process.



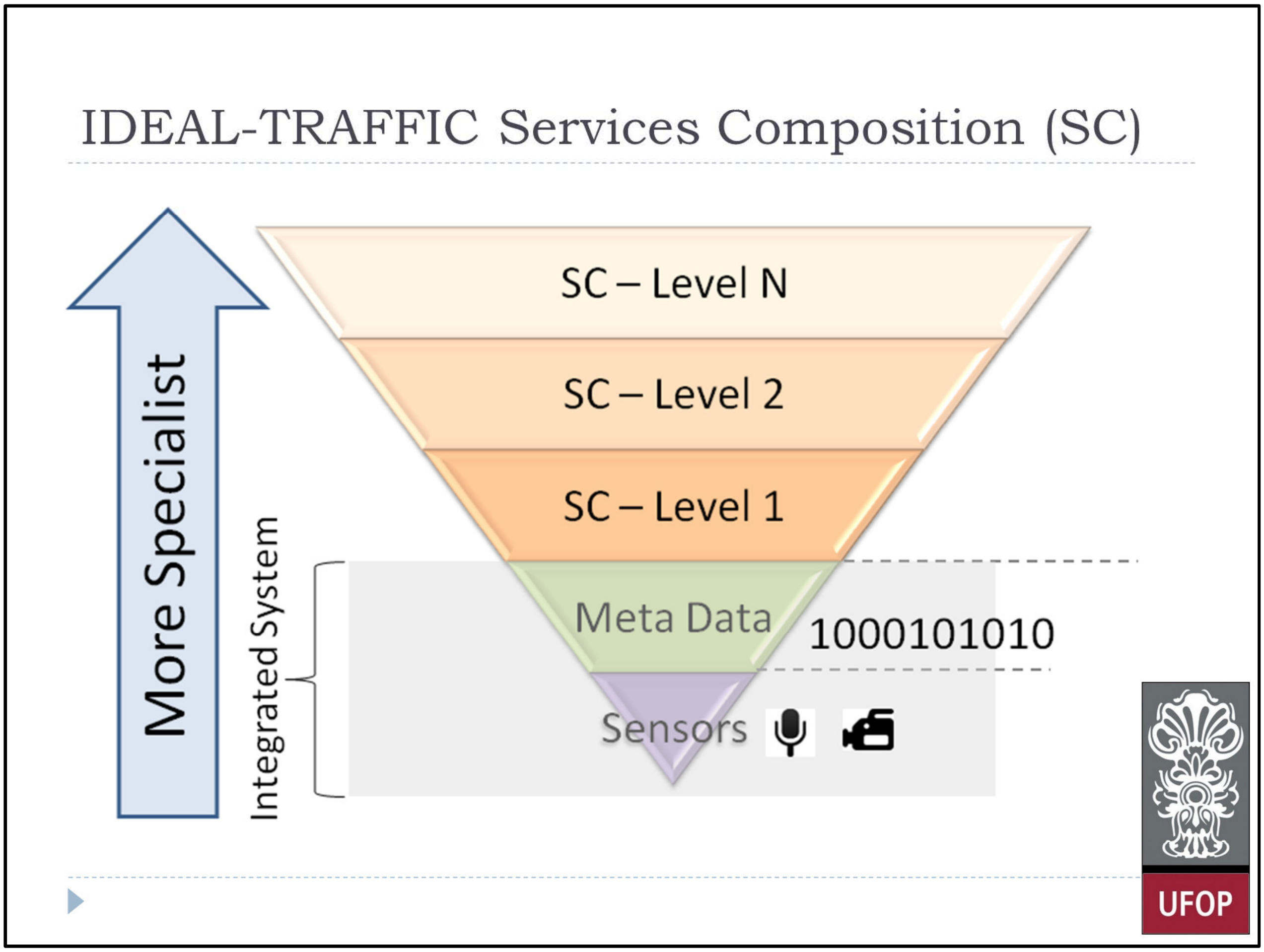
Adaptability: Adaptation can ensure the following requisites: Fault Tolerance of system (whenever there is element failure); the increase of quality of service (QoS) delivery, and consequently the increase of service level agreement (SLA); the promotion of mechanisms that can save energy in the presence of use embedded architecture, and lastly, the betterment of the use of the resources.

Heterogeneity: A system is heterogeneous when it runs in different hardware (embedded, PC, Server, Workstation), with different software (Android, Windows, Linux) and is available in several communication networks (Ethernet, WiFi, UTMS).

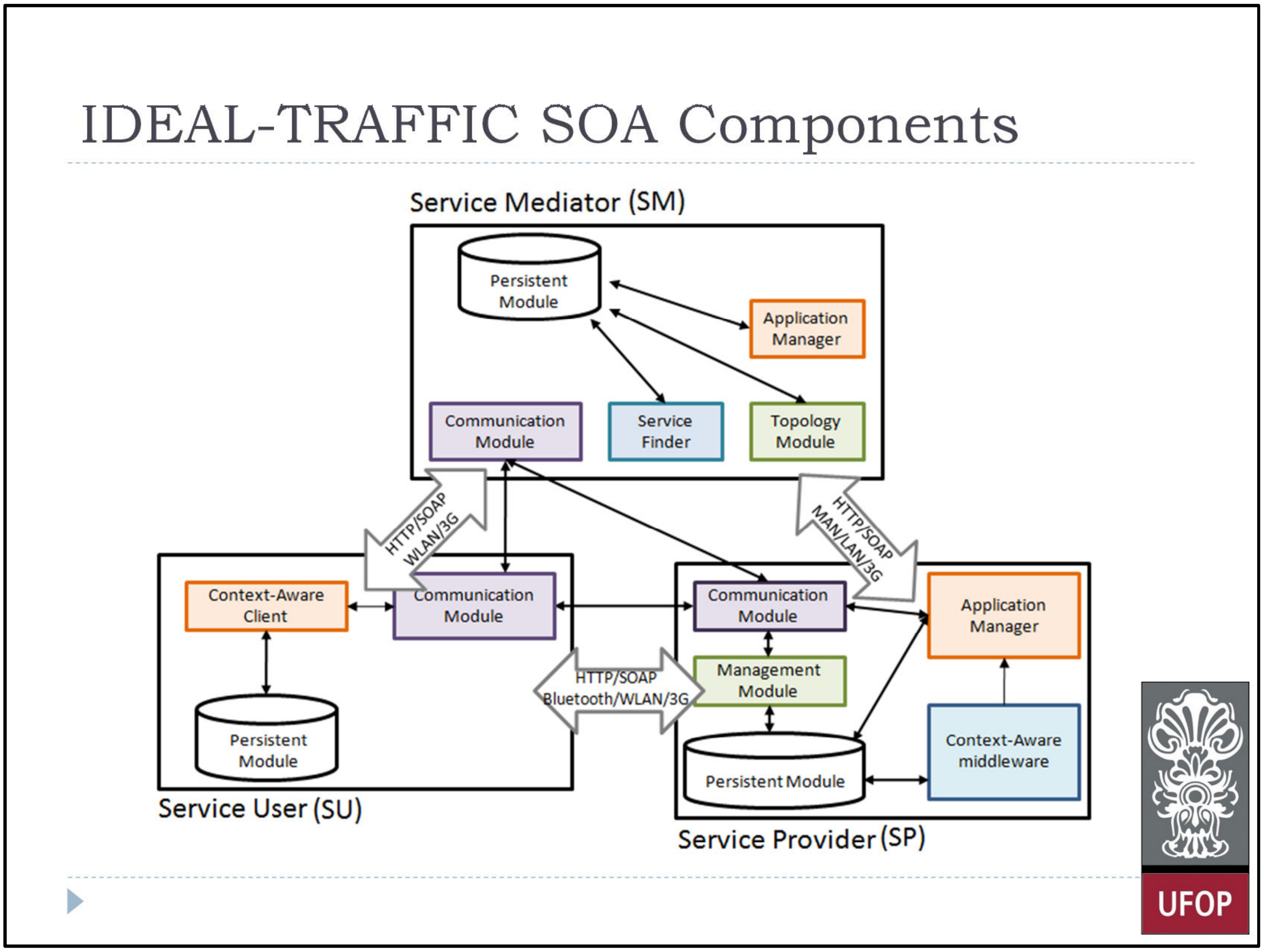
Interoperability: The system is able to establish connection with other components with regard to creating monitoring solutions, it is common to find legacy systems that may be used together with new technologies. In these cases, the legacy system must have an interface that is able to connect to the interface available in the IDEAL-TRAFFIC.

Scalability: There are three different scalable ways to measure distributed systems. First, the size of the distributed system. Second, the geographical size when the users and resources are distant. Third, the administrative scalability means that the distributed systems continue to be easily manageable regardless of their growth.

Openness: This requirement is made out of three sub elements: 1) Each service is accessible to all users; 2) (Registry) applications are easy to implement, install and deploy; 3) The stakeholders can write their own applications (in this case, the system administrator).

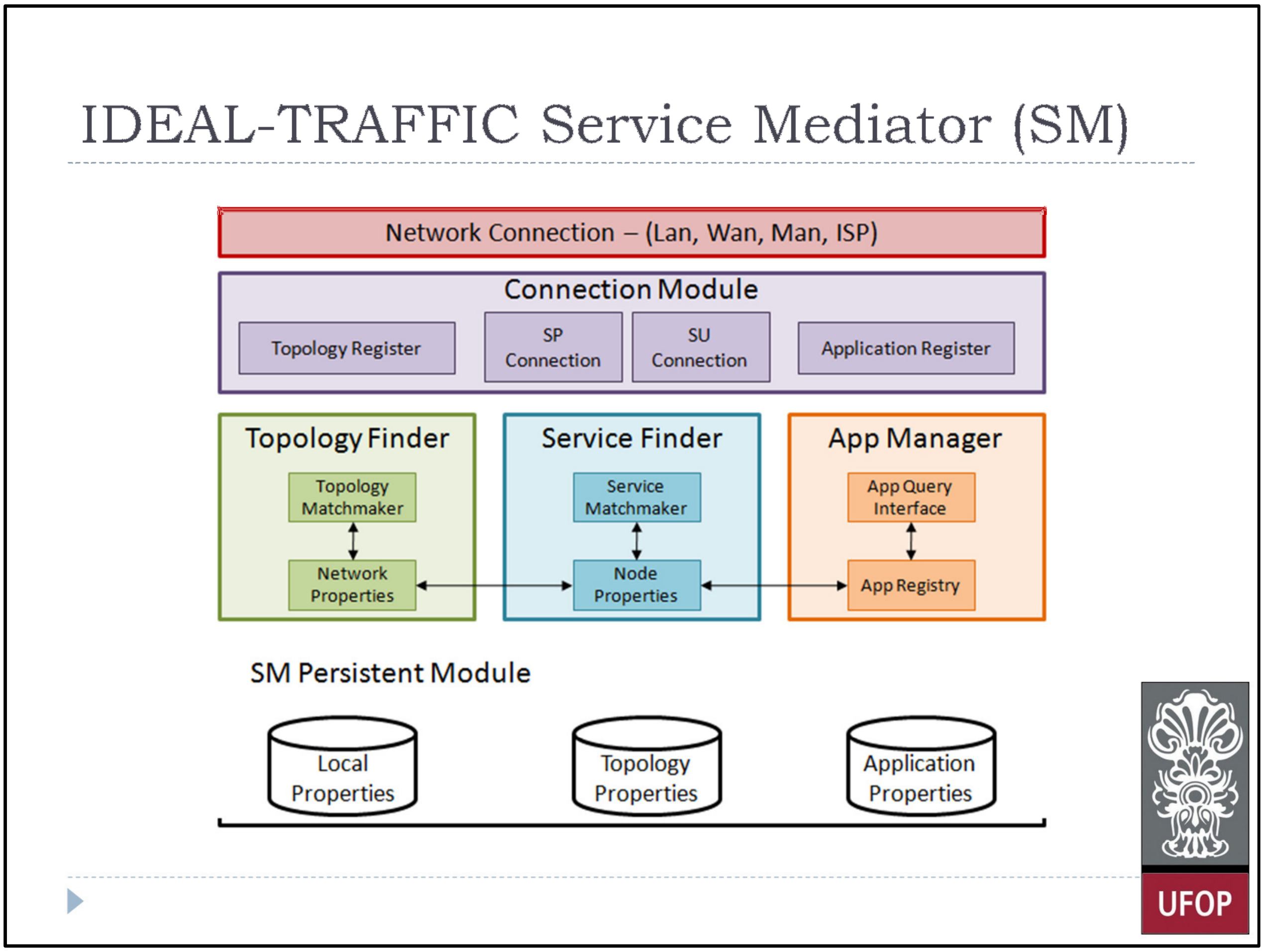


Three elements constitute IDEAL-TRAFFIC: Service Mediator (SM), Service User (SU) and Service Provider (SP). The SP element is the main component to perform an SC hierarchy. The higher the level, the more specialized the rules will be. This slide shows the SC hierarchy. The two first levels are atomic services. Programs such as digital image processing, or signal processing, perform these services. For instance, when the sensor is connected to a camera or microphone respectively. The result of atomic services can be combined with other services, which may be provided by the same SP, or by others. In our architecture, it was defined that the two first hierarch levels are integrated on the same SP.



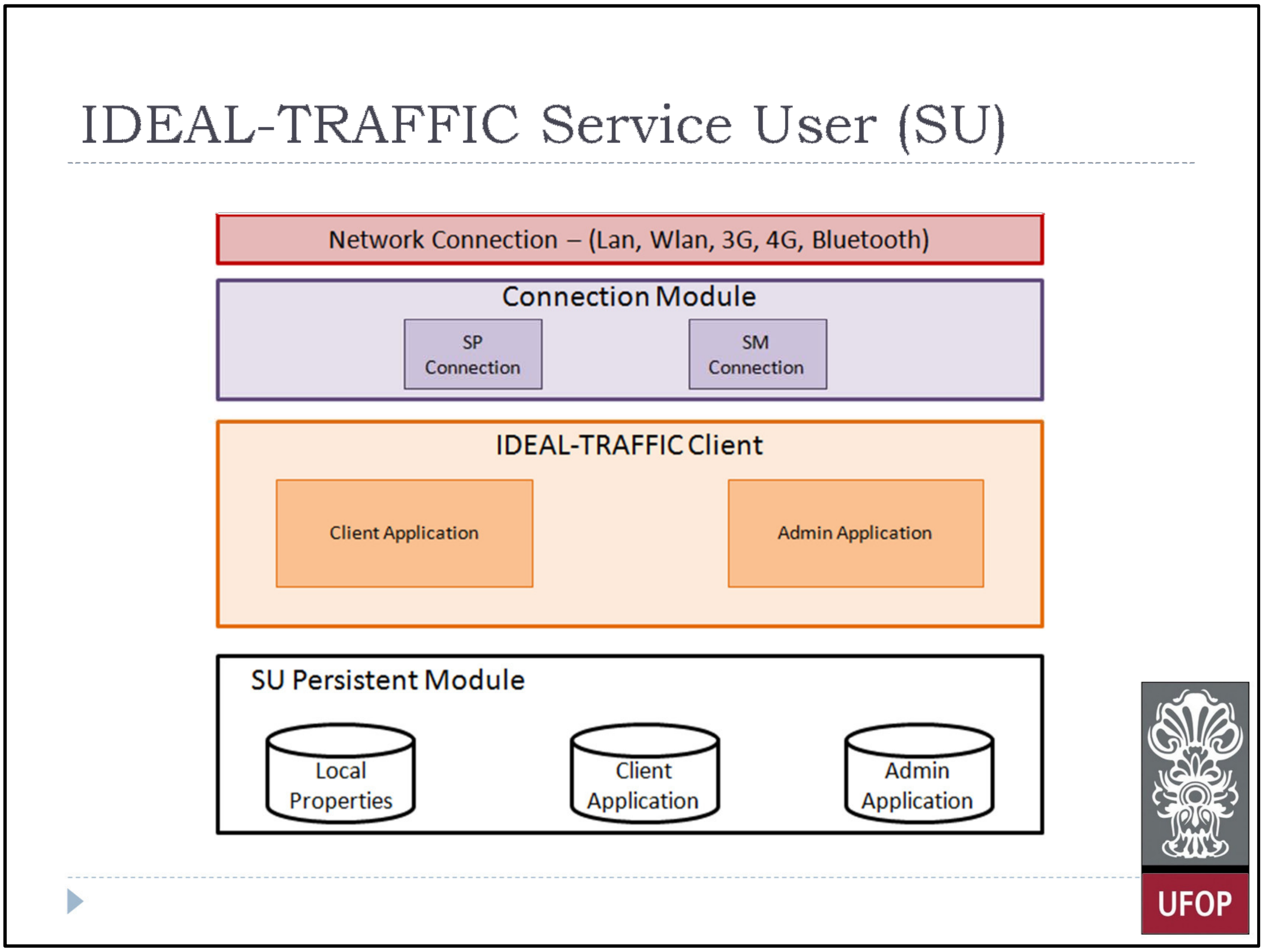
Communication among the three services may be performed by several network interfaces. In general, SUs are mobile devices that can found in the market today, such as tablets and smartphones. These devices are created to work with a diverse set of wireless technologies such as WIFI, Bluetooth, 2G, 3G and more recently 4G. It is also important to take under consideration stationary devices, such as PCs, which may need cable, or Lan, connections. Following the same pattern, SPs can be built in embedded platform to work with the same interfaces, cabled or not. Communication between the SU and the SP is evident since both share the same technology. Usually the SM is allocated on data center, and has a more restrict wireless technology. These devices work over LAN, WAN and MAN networks.

The services can establish communication using the HTTP (Hypertext Transfer Protocol), SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language), and UDDI (Universal Description, Discovery and Integration).

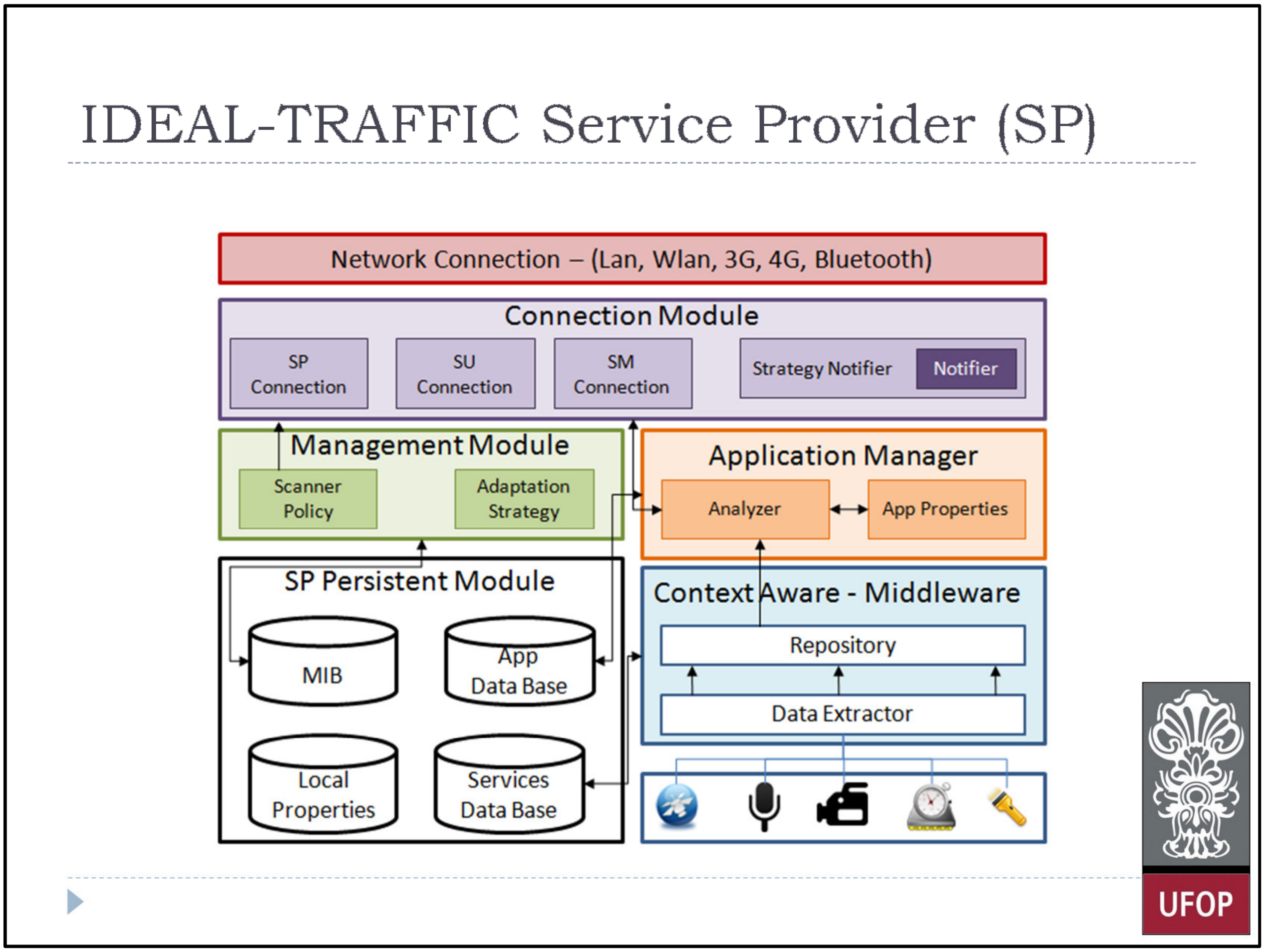


The IDEAL-TRAFFIC SM is a kind of central element. It mediates the communication between SP and SU, and it manages the status of network and state of the SC applications. These components are responsible for analyzing the request and delivery of the correct SU data according to the type of request being made. Requests that are related to administrative tasks are queried in the SM database and forwarded to the SU. The SM manages the network topology and discovers services on the SPs. This information is delivered to the SU administrator interface.

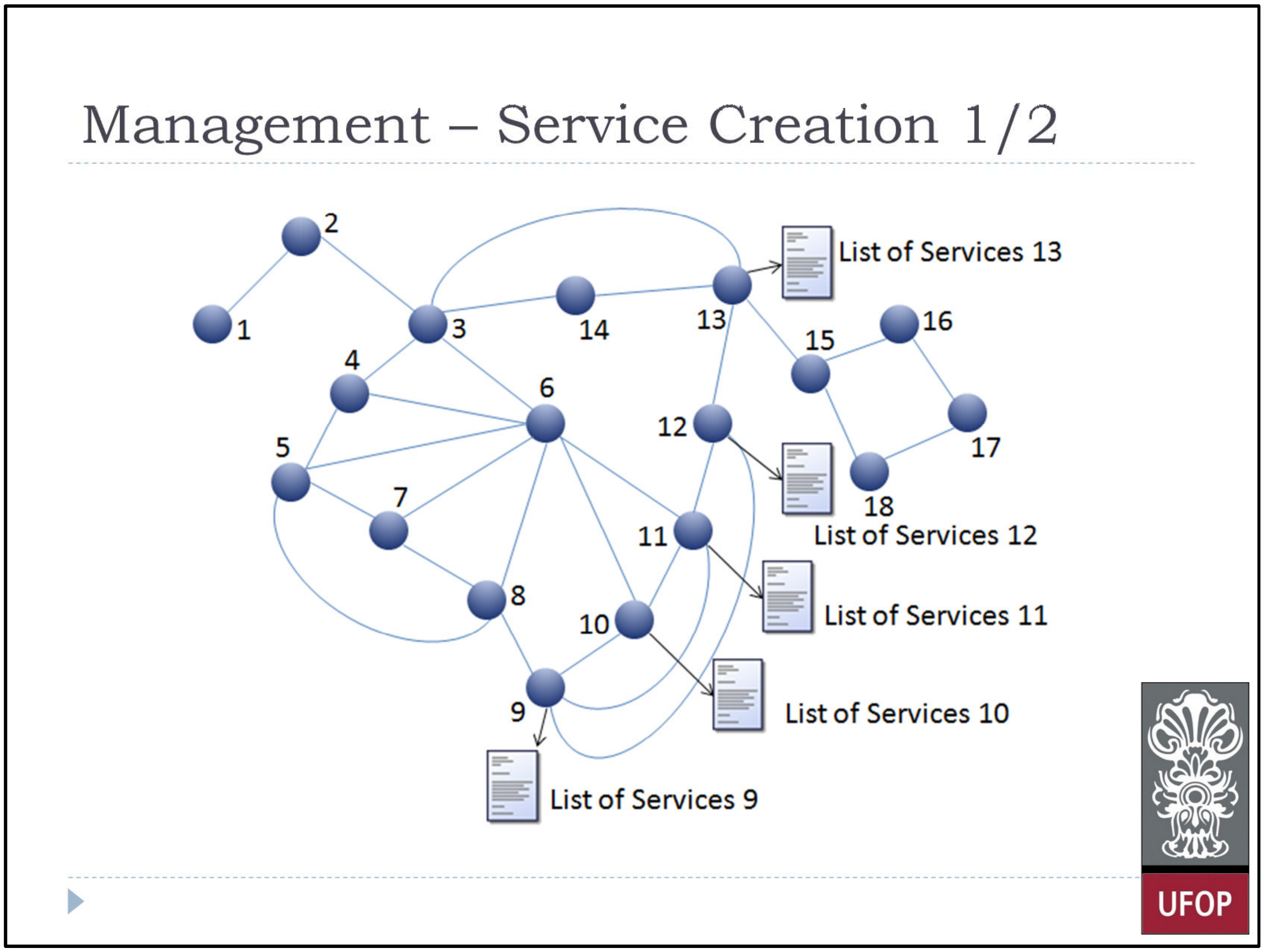
For a end-user request, the SM queries the properties of available SC and sends this information to the SU interface. After the end-user has chosen the service, the SM binds its SU in the correspondent SP.



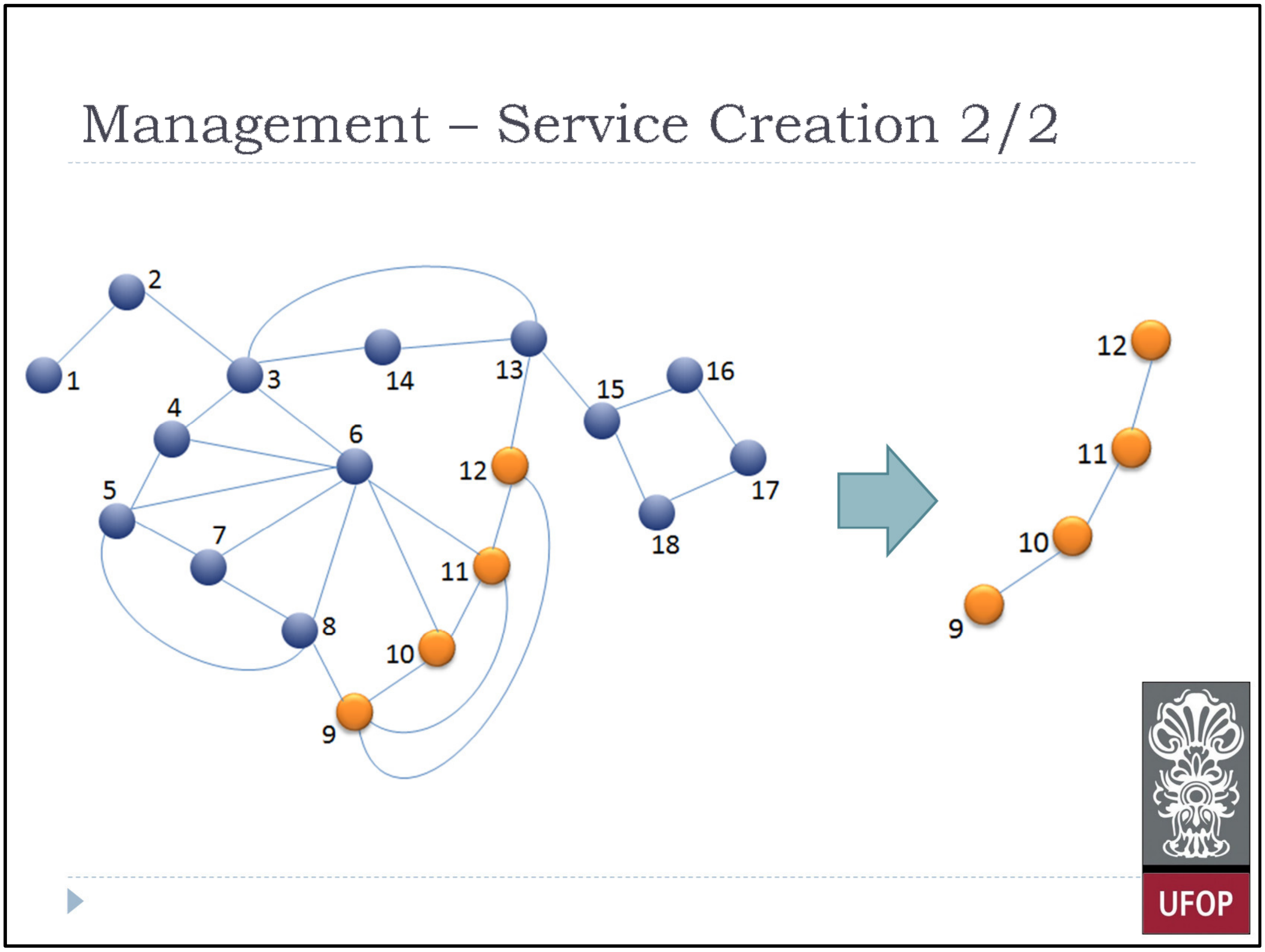
IDEAL-TRAFFIC SU corresponds to an interface to user request in the SOA architecture. This component is composed by a friendly human interface which establishes connection with the services and provides data to the users. Users typically define mobile devices like smartphones and tablets, as favorite hardware to carry out the communication. The communication process happens through the SU request made for the SM. The latter treats the request and retrieves the related information by user. Two interfaces are proposed in the IDEAL-TRAFFIC, 1) An user interface to manage context-aware applications. 2) An user interface to request the services available for users. The former is dedicated to system administrators, while the latter is for final users who benefit from the several services available on SPs.



The IDEAL-TRAFFIC SP is the main element to compose the SC application. The component usually is linked to sensors and knows programs to extract information, or composes the data received by other SP component. After SP receives the message of registry in a SC, SP often scan its parents, and in the case of discovering some failure, the SP starts the adaptation process. The modules that perform SP this role are described below.

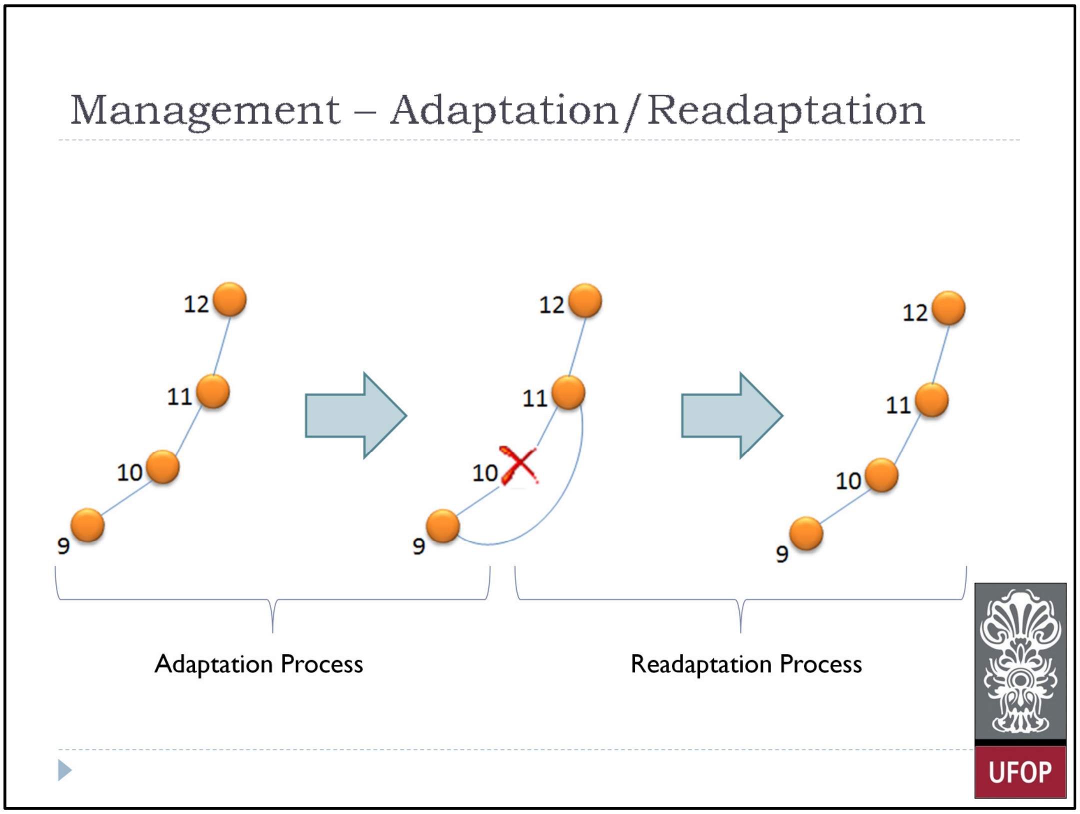


A logged system administrator calls the function to create an application in the SU. At this point, the SU has retrieved the topology of network formed by SPs. The system administrator then selects possible candidates with services in order to form a SC. Remember that each SC has autonomy to change its list according to its state. The Figure on slide shows this process. The elements having a List of Services attached to them correspond to the possible candidates selected by the system administrator. The others SPs are SCs available to use at that moment.



In this step, system administrators have already chosen the SPs that will participate to the SC. In the case the SP 13 has not yet been chosen, since the service need to perform SC is not available on it. The 9 until 11 SCs will participate of the SC. Note that each SC selected has connection with more than one peer. System administrator chooses how each node will establish connection to form a new sub graph.

The system administrator configure final rules and properties in the SC, then validates it and sends it to SM to be registered. The registry is done in the SM, which saves the general information of the SC application. SM generates a SC identifier (id) and sends it to each SP participant. The SPs register the id together with the SC properties.

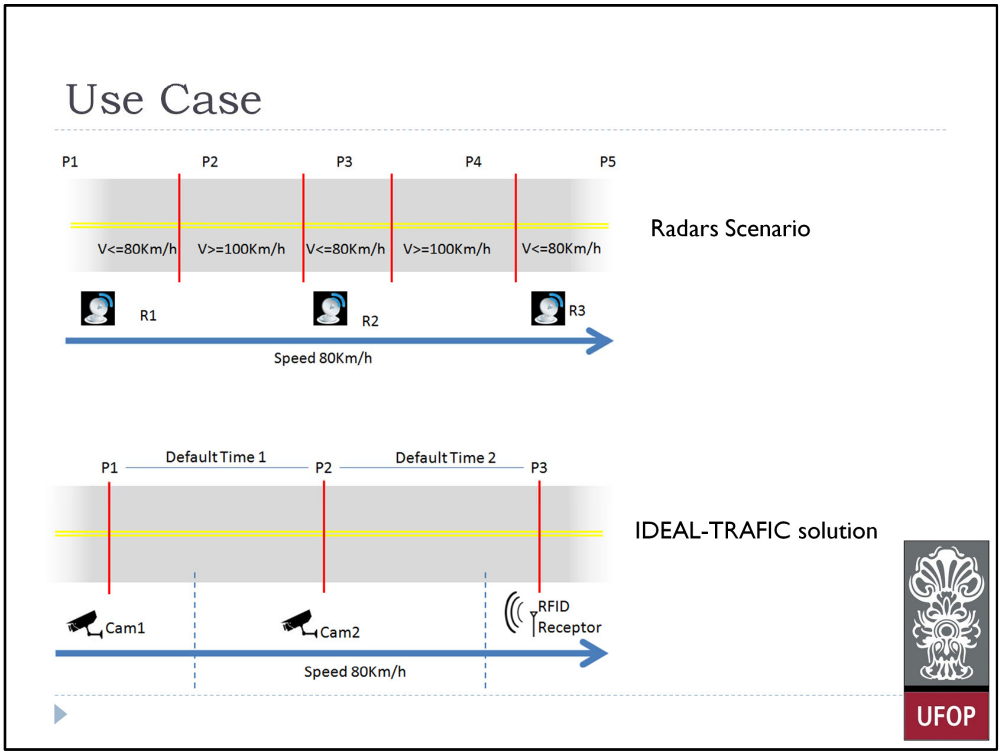


Adaptation:

For instance, it assumes that the SP 9 sends its results to SP 10; that the SP 10 sends them to the SP 11 and that the SP 11 sends the results to the SP 12. Then, by definition the SP 10 is named "parent" of the SP 9. In the IDEALTRAFFIC framework, all SPs "children" are responsible for verifying the connection with their "parents". We use this definition because if a parent stops working, the "child" tries to find another "parent".
In our scenario the SP 9 realized that the SP 10 had turned off by means of the Scanner Policy component. Consequently, the SP 9 sent a broadcast message to identify candidates that could assume SP 10's place. Note that, SPs 8, 11 and 12 are candidates. In our explanation, due to a requirement, SP 8 cannot perform SP 10's role. SP 11 was chosen because it was available and it was also the first in the priority queue. The new connection is shown. After establishing new connections, a new SC topology is registered on the SM.

Readaptation:
In the case that SP10 may resume its activities, a process will be initiated as an attempt to readapt the component. The faulty component, namely SP10 in this case, is responsible for the management of all the procedures involved in the readaptation. SP10 will then need to establish a connection with the SM, seen that it is aware of the current status of the application which the SP10 held before it had gone off-line. If the application remains active, the readaptation procedure will then take place.

After booting, the SP 10 collects in its database the information of its last state. Information on the SC(s), in which SP 10 was a participant, is retrieved. As a first step, SP 10 sends to SM a message to confirm if the retrieved SC is still registered in SM and if it is authorized to start the Readaption process. If positive, the process continues, otherwise it stops. Upon receipt of the authorization, SP 10 starts a procedure of communication with its peers. In this case, SP 9 and SP 11. Both peers were retrieved from SP 10 database. SP 10 then, multicast a message to its peers inviting them to re-establish SC participation. If SP 9 and SP 11 responds by demonstrating that they are able to receive SP 10 in the application, SP 10 will build the necessary services and synchronize them with SP 9 and SP 10. SP 9 and SP 11 receive the message, and communicate the changes in SC topology with their current peers. The SC topology is registered on the SM and each SP carries out the necessary registries in its database.

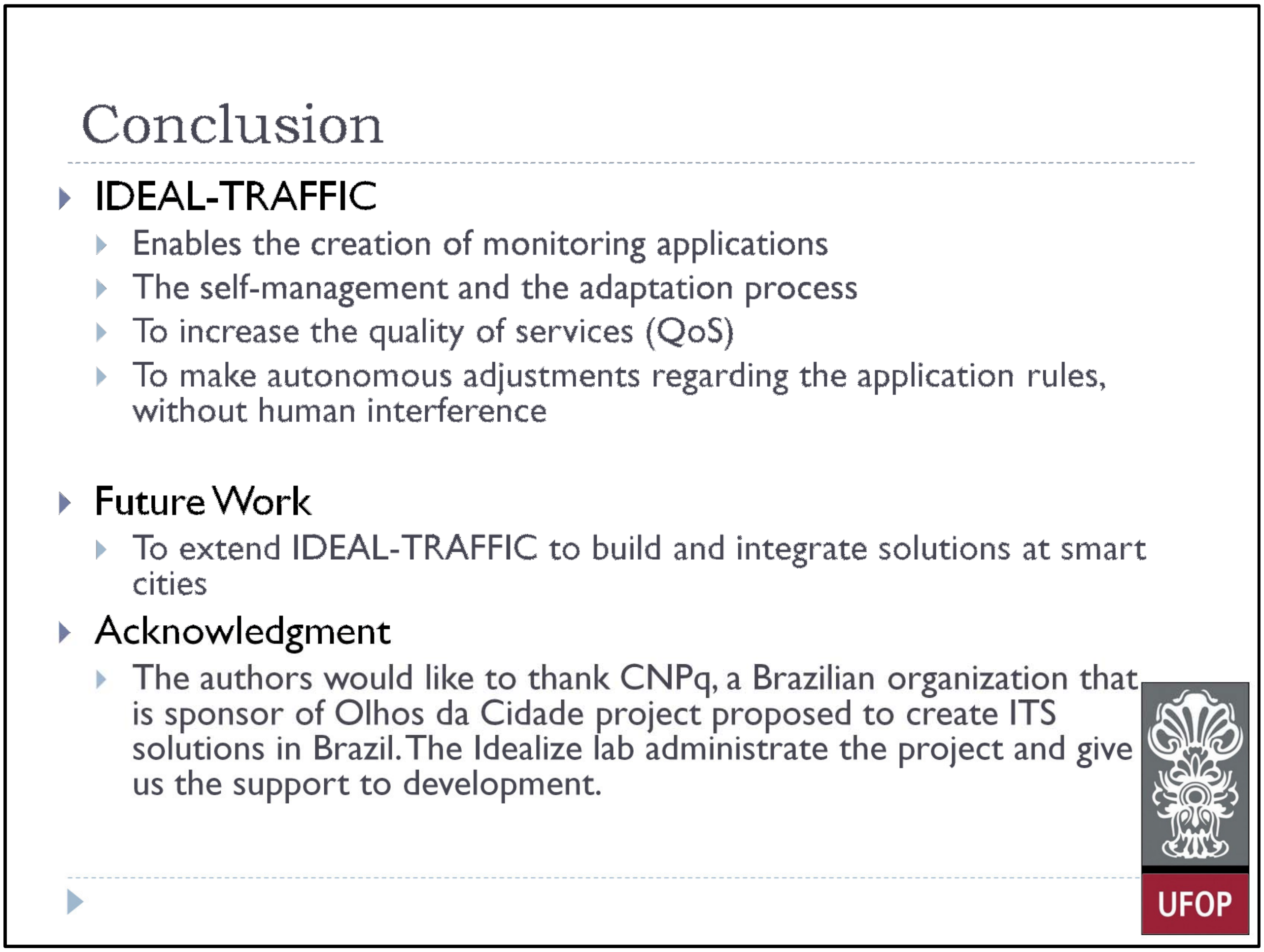


Radars Scenario: The road layout was divided into five parts (P1, P2, P3, P4 and P5). The vertical lines between each part Pi delimit the area which have P1, P3 and P5 has R1, R2 and R3 radars respectively. The average speed in this scenario is 80Km/h. The radars cannot measure and ensure the average speed imposed by the traffic administrator. This is due to the fact that this device can retrieve only an exact speed at that point. The driver's behavior is usually similar to what is shown on Figure Radars Scenario. In those cases, the radar points have the defined speed S, and in the areas without radar, the speed S increases.

We are encouraged to solve this problem because the frontier line between points P2 and P3, P4 and P5 have high collision risk. In these areas, drivers tend to dramatically reduce the speed, after seeing radar signs on the road. An inattentive driver coming up from behind can have difficulty avoiding collision.

IDEAL-TRAFIC solution: Each radar point has been changed by collectors that identify vehicles, where P1 and P2 were chosen to use cameras and P3 was chosen to use a RFID antenna. The first function of P1 and P2 is to obtain the tag number of a vehicle on road. P3 extracts a RFID tag identifier of the vehicle. In a real scenario, the P3 collector may be a legacy system which had been linked with the IDEAL-TRAFFIC instance, thus evidencing another benefit of IDEAL-TRAFFIC. It is assumed that when a vehicle exceeds the average speed, the SC registers an event.

The role of P1 IDEAL-TRAFFIC SP is to extract the tag number, from the image collected by the camera device along with the current time. Both information would be packed and sent to P2 SP, where they would be combined with the same information obtained from the P2 camera. The algorithm combines both information, from P1 and P2, using the Average Speed Formula. If the average speed analyzed by P2 is greater than 80km/h, then P2 generates an event. Note that this is the SC application rule. If by any chance, P1 loses connection with P2, P1 would then adapt the topology connection with P3. If this happens, the properties used in average formula will change, such as the distance between P1 and P3.



This essay presented IDEAL-TRAFFIC: a framework based on the SOA architecture that enables the creation of monitoring applications. As it was seen, the self-management and the adaptation process of IDEAL-TRAFFIC have the ability to increase the quality of services (QoS). This is due to the fact that the service composition is able to make autonomous adjustments regarding the application rules, without human interference. This essay also presented a use case that consolidates the framework by using an ITS application, demonstrating as a result, how the IDEAL-TRAFFIC can be applied in traffic solutions.

In a future work we will suggest that the use of the IDEAL-TRAFFIC be extended to build and integrate solutions at smart cities.

References

- J. Rubert. (2007) Service oriented architecture standards. [Online]. Available: http://www.w3.org/2007/01/wos-papers/boeing
- D. Booth, H. Haas, and F. McCabe. (2004) Web services architecture. [Online]. Available: http://www.w3.org/TR/ws-arch/
- G. K. Don Box, David Ehnebuske. (2000) Simple object access protocol (soap) 1.1. [Online]. Available: http://www.w3.org/TR/2000/NOTESOAP- 20000508/
- E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. (2001) Web services description language (wsdl) 1.1. [Online]. Available: http://www.w3.org/TR/wsdl
- L. Clement, A. Hately, C. von Riegen, and T. Rogers. (2004) Uddi version 3.0.2. [Online]. Available: http://uddi.org/pubs/uddi-v3.0.2-20041019.htm
- C. Ahn and Y. Nah, "Design of location-based web service framework for context-aware applications in ubiquitous environments," in Sensor
- Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010 IEEE International Conference on, june 2010, pp. 426 –433.
- A. Staikopoulos, O. Cliffe, R. Popescu, J. Padget, and S. Clarke, "Template-based adaptation of semantic web services with model-driven
- engineering," Services Computing, IEEE Transactions on, vol. 3, no. 2, pp. 116 –130, april-june 2010.
- A. Shimura, T. Sakaibara, M. Hiraiwa, and T. Aizono, "Proposal of an autonomous group-management model and its application to intelligent transport system," in Autonomous Decentralized Systems, 2003. ISADS 2003. The Sixth International Symposium on, april 2003, pp. 71 – 79.
- M. Padmadas, K. Nallaperumal, V. Mualidharan, and P. Ravikumar, "A deployable architecture of intelligent transportation system - 2014; a developing country perspective," in Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International Conference on, dec. 2010, pp. 1 –6.

UFOP