Adaptive Communication Agent for Group Communication activities

Sakkaravarthi Ramanathan, Khalil Drira, Christophe Chassot, Thierry Desprats, Sibilla Michelle

Adaptive Communication Agent for Group Communication activities

Sakkaravarthi Ramanathan[(1)], Christophe Chassot[(1,2)], Thierry Desprats[(2,3)], Khalil Drira[(1,2)], Michelle Sibilla[(2,3)],

[1] *CNRS ; LAAS ; 7 avenue du Colonel Roche, F-31077 Toulouse, France*
[2] *Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France*
[3] *IRIT, Paul Sabatier University, Toulouse, France*

In group-wide collaborative environment, managing the adaptive communication is a challenging task. It implies the monitoring of sudden change in the activities while providing a solution to maintain the connection with the available resources. In this work, we design a software agent that supports autonomic computing to ensure reliable communications among the mobile devices and Autonomous Ground vehicles (AAV). This issue is addressed in the context of save and rescue missions carried out during natural disasters such as floods and forest fires by human and voluntary operators within the framework of wireless environment. The paper focuses the autonomic functionalities of the components used to monitor, analyze, plan, and execute the adaptive mechanisms in case of evolution (mission/environment). We distribute this agent among devices and vehicles to ensure the adaptive task and it is tackled by using appropriate policies used to select the decision and executed without manual intervention. This research is applied to a Crisis Management System (CMS) within the context of the French RTRA project (ROSACE).

Keywords: Communication, Software Agent, Autonomic computing

1. Introduction

The evolution of mobile devices provides new opportunities for the development of systems supporting group-wide collaborative application over wireless networks. Such systems allow a distributed group members to collaborate and share a common task that composed of various resources and applications. Our scenario involves mobile actors, ground and aerial communicating robots along with human volunteers using mobile devices operate within the wireless communication context to save and rescue people in case of natural disasters such as forest fires, flooding, etc.

Many existing work have contributed to the control management within collaborative environments for the wired network. However, in a mobile environment, which is realized by the wireless access networks and use of smart phones, the mission requirements vary dynamically and continuously. Even the availability of access network cannot guaranty the reliable communication due to packet loss, latency and bandwidth. Hence we need an autonomous computing to overcome the challenges due to mission evolution. Several work deals with the context awareness, most of them suggested an adapted approach based on a centralized architecture whereas we are concentrating on the distributed one.

The system is concerned of establishing a new collaborative communication session and maintaining the reliability of a created session. In case of unpredictable situation (mobility of actors, limited energy resources), system execute adaptive mechanisms by monitoring the data flow. Also, the actor parameters such as role and availability may influence the present conditions of the communication. Here, the devices used are characterized by their limited capabilities (low battery power) and access networks do not guaranty the reliable bandwidth because of frequent connection fluctuations. So, a mechanism needs to sense the changing environment and based on this change, system decides and executes an appropriate adaptation. The communication devices equipped with several network interfaces (WiFi, GSM) need to have an intelligent module that shifts the access technology according to the availability.

To face these challenges, we have designed an autonomous software agent which is distributed to collaborate with other agents in order to detect the mission requirements like establishing local network among the actors, searching victim, locating victim's position, etc. In order to achieve various task, agent executes adaptive decisions in hierarchical fashion, i.e., trying to solve the problems locally and in case of no solution, it triggers other components/agents for collaborative actions.

This paper is structured as follows. Section 2 presents related work and section 3 outlines our ROSACE scenario with brief challenges and requirements. Section 4 details the design of our communication agent (CA) and also the collaborative deployment strategy and context management technique. Section 5 focuses on operating scenario explaining the implementation of our work. Section 6 concludes the work with open issues.

2. Related work

In [1], an adaptive framework supporting multiple classes of multimedia services with different quality requirements in wireless cellular networks is proposed and focused on adaptive policies at communication level. The work in [2] envisions middleware architecture for service adaptation based on network awareness to manage resources in an adaptive context. Though it highlights the necessary adaptation, there is no solution for context aware problems. Further research [3] provides frameworks for designing transport protocols whose internal structure can be modified according to the application requirements and network constraints. In [4], a schema is described for dynamically managing distributed computing resources by continuously computing and assessing quality. Here, resource utilization metrics are determined a posteriori and adaptive distributed system reference architecture is equally put forward.

The context adaptation is the set of rules which control and anticipate the changes that may occur in the environment in order to provide an appropriate service [5,6,7]. User defined rules intuitively enumerate a set of adaptation rules in order to meet the challenges. In case of mobile environment, the predictive capacity of the scenario is almost depends on various parameters. For example, actor's location, available access network, bandwidth, energy level of the devices play a major rule for the autonomous computing to select the predefined policies. Earlier works have used the agent paradigm, but collaboration management has not been addressed in case of mobile environment.

Motivated by the above discussion, we present a distributed agent for reconfigurable adaptive communication system. Our framework retains the MAPEK [8] technology for autonomic computing and the agent communicates to internal component of the devices and/or to other device'agent for solving a problem. In our work, the distributed software agent shares the autonomic management by properly executing the adaptive policies. Constraint in energy resources may not support the collaborative sessions and this will be handled by shifting and switching the network access technology, thanks to the agent that collects the context information and takes decisions autonomously.

3. Scenario and challenges

The three major actors in our scenario are mission supervisor, coordinators and investigators. The supervisor's function is to monitor, manage, decide and authorize actions to coordinators and investigators. Coordinator's task is to report to the supervisor and to manage the investigators during the mission and assign tasks. The investigator's role is to explore the operational field, observe, analyze, report about the situation and help the victims. To support this, we have coordination and cooperation flows. Coordination flows take place between investigators and their coordinator and between the coordinators and the supervisor. Cooperation flows occur between the investigators within the same group or between the investigators of different groups.

Figure 1 describes the ROSACE scenario with the flows and network connection. The two trucks (coordinators) have the WiFi access points to communicate with the investigators like robots, firemen. The control center connects a coordinator using satellite and another one using the WiFi access point. But if there is a network problem between the control center and a coordinator, one way to solve the problem is to contact the other coordinator, then its investigator to reach the lost coordinator. Another way is to create a network, thanks to the walker's mobile devices; communication can be established as shown in the diagram.

When a natural disaster happens, the control center is responsible to make an action and a team is sent to that area to provide emergency assistance for victims. The team tries to stabilize the situation and reduce the probability of secondary damage and speed recovery action. Before this process, the control center must gather information about the site map, a list of the important objects at the site, and some previous reports and materials obtained by organizations like Telecommunication industry, public and private organizations, etc. It acts as back-end (high power computer integrated with data, knowledge and content), providing advanced services to ROSACE front-end devices held by team operators. The team forms a mobile ad hoc network in which the team leader's device coordinates with other member's devices by providing appropriate information and assigning activities.

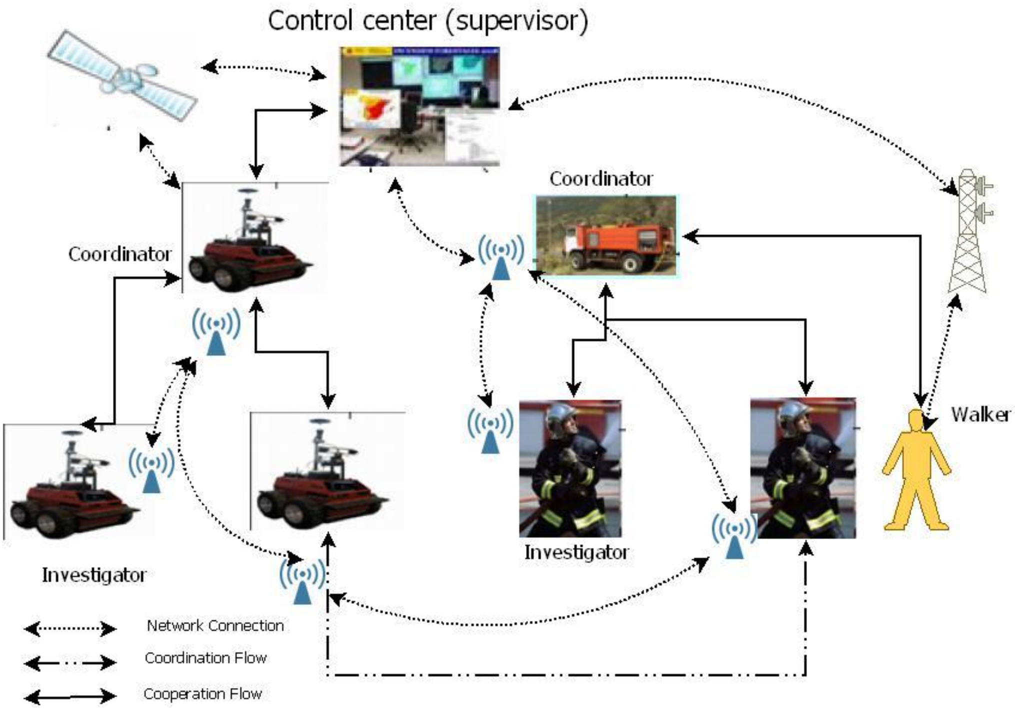

Figure 1. ROSACE scenario description

Challenges

The challenge is to guarantee the permanent connectivity among actors and to offer best-possible quality of communications according to mission goals and available resources. The system has to monitor, detect the changing environment and adapt to the situation and if there is no solution to rectify the problem, it has to process the request to its peer then to control center. To support this schema, network-oriented services provide ubiquitous access to peers and be technically transparent, taking into account different context requirements depending on the targeted activity, user' mobility, exchanged data flows (audio, video), and constraints such as variable communication and device resources.

Finding a mechanism to ensure context-aware behavior for the process of monitoring is quite challenging. It can be reached by using a set of predefined policies that dynamically control by considering the current network status and the configuration of monitoring functionalities. The communication objectives of ROSACE system can be designed by combining both classical and autonomic management mechanisms.

The context adaptation anticipates the context change that may occur in the environment but it is impossible to enumerate all the possible conditions, hence diversity of solutions can be provided according to the collected parameters.

The aim of ROSACE scenario is to save the victim and to do so, the software agent installed in those devices should be aware of mission information. Environmental context has to be monitored and collaborated with other nodes for participating in service provisioning. To achieve adaptive communication needs, agent has to manage internal service APIs, asses' service quality according to mission needs and to perform dynamic re-configuration in order to satisfy telecom service requirements. This includes a) achieving its own goals by sharing common resources with other agents, b) managing internal resources to satisfy collaboration requests, c) interacting with other agents to exchange control information and data to guaranty end to end service provisioning.

The basic requirements to achieve network connection are:

• Each device should include hardware to know its communication distance from the surrounding devices that are within radio range. Specific techniques and methods are easily available, i.e., TDOA (time difference of arrival) and SNR (signal-to-noise ratio).

• Each device is equipped with GPS hardware.

• At start-up, all devices are connected (that is, each device has a path to any other device). Each device doesn't have to be within range of any other device, but it requires at least a loose connection, guaranteed by appropriate routing protocols.

• The coordinator predicts disconnections and manages its members for reassignment of the process tasks.

4. Communication Agent Architecture

We use Unified Modeling Language (UML) to specify, visualize, modify, construct the ROSACE framework as it offers a standard way to visualize a system's architectural blueprints, including elements such as, actors, activities, etc. For example, the diagram 2 describes the use cases of coordinators ‘functionalities.

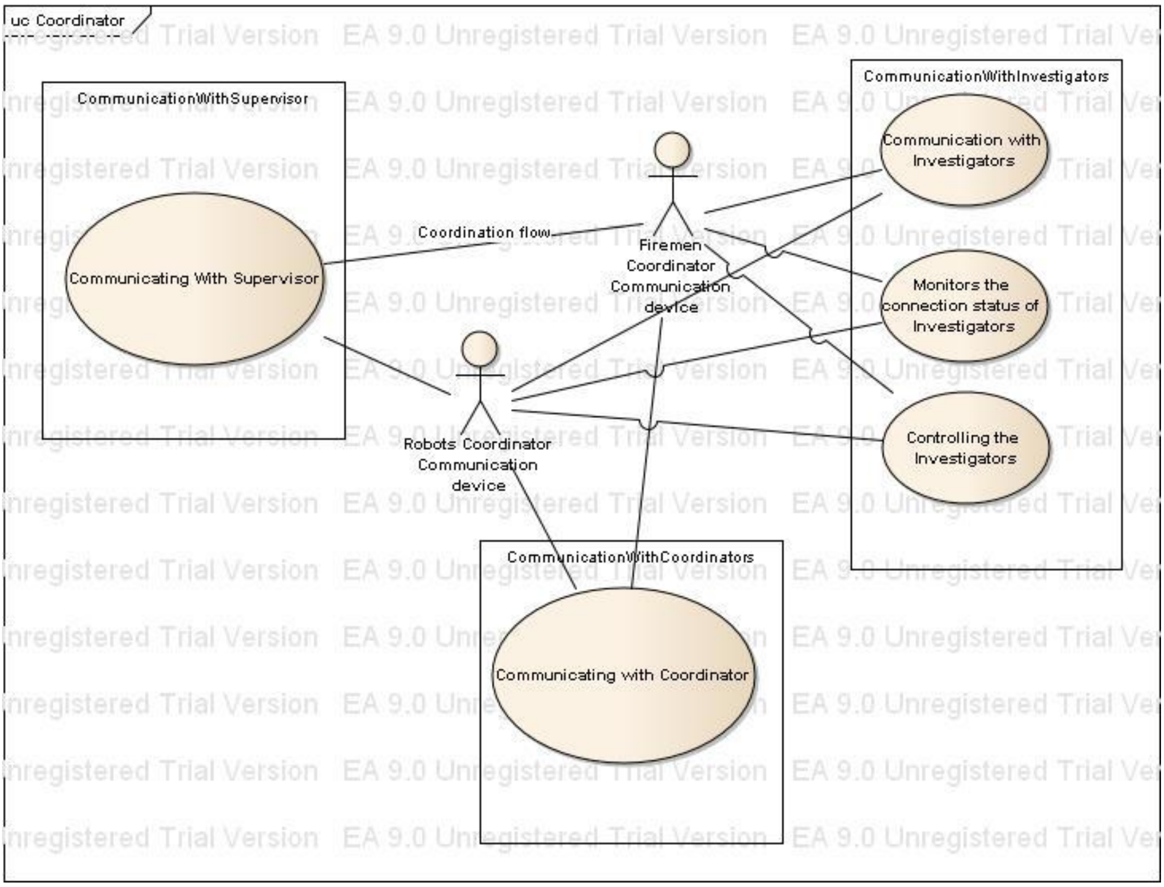

Figure 2: Use cases for Coordinator’s functionalities

To assure connectivity, the devices are embedded with multiple network interface card. For example, the smart phones owned by the firemen have two network cards. One is dedicated to standard WiFi network and another one for GSM. The WiFi card has the capability to switch between infrastructure to adhoc mode depending upon the availability and it is activated by the autonomic computing which will be discussed later. Even though it has two network cards, only one card will be activated at any time as it is energy constraint terminal. But in case of robots, power is not a major problem and thus installed with three network interface cards. It allows us to activate two cards at same time to connect to two different actors through different access networks to assure the availability.

The state diagram in figure 3 explains the adaptive mechanism to create the local network at the intervention area. Once the devices are launched, the network card searches for access point to get connected. In our hypothesis, our priority is to have the WiFi IFM at the first place. Once the device has the signal from the access point, it will get connected automatically. If not, our autonomic computing switches the device to adhoc mode to detect any neighbor in same mode. If yes, it will connect to the neighbor and still there is no signal from the adhoc, then it activates the GSM to get connected to the supervisor. Until it connects with any of the ROSACE members, the autonomic module searches for the connection which is the state 0.

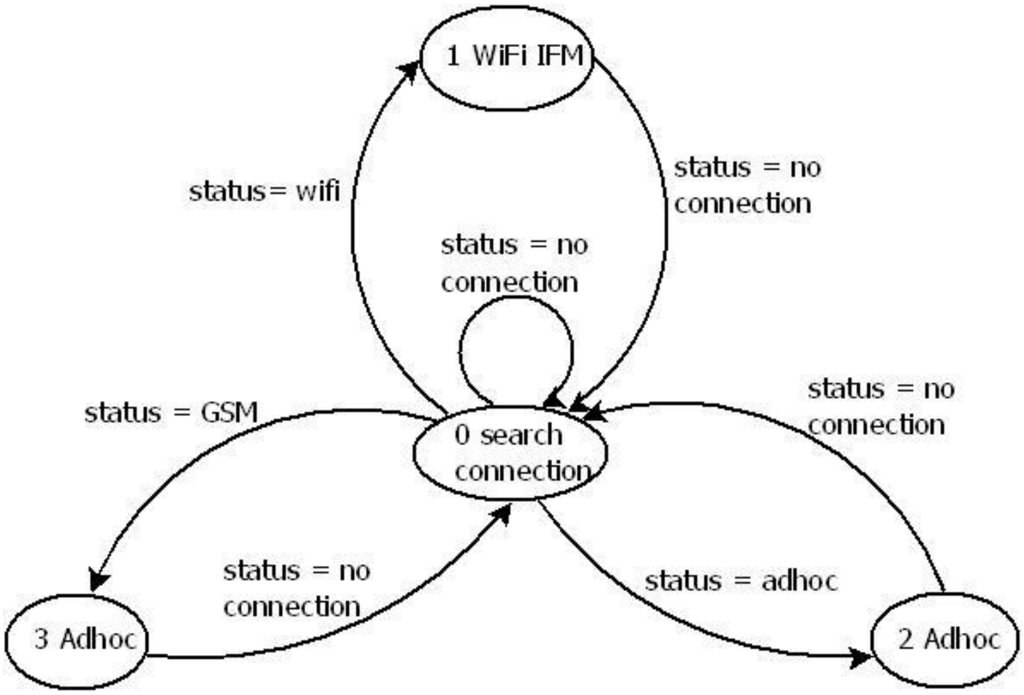

Figure 3: State diagram for adaptive mechanism for creating local network

4.1 Communication Agent

It is a software component installed in ROSACE devices used to detect the context parameters and take an adaptive decision in case of sudden change in the mission. The location of the actor is detected by using GPS, an application pre stored with the device. The connectivity and the availability parameters are determined from the autonomic computing. CA consists of many internal components monitoring is done at different network layer. The detailed composite diagram of CA is shown in figure 4. Each component serves its specific purpose and it composes MAPEK technology for autonomic computing. In case of Wifi network connection manager, it checks the radio connection and MAC address of the device. If there is any failure, it informs to the communication Node manager. These internal modules invoke each other to communicate with each other, once the plan executes the appropriate policy.

Communication Node Manager

This manager monitors and collects overall network configuration and its context information (networkID, connected member list, access interfaces –WiFi, GSM, etc.). It interacts with the knowledge database in case of problem to identify the causes and finally execute adaptive decisions. This manager has information about configuration parameters, power consumption, and operation status of the devices.

Communication Service Manager

This manager monitors the application status and negotiates the service requirement in case of demand in the service. Priority handling and message content adapting are handled here.

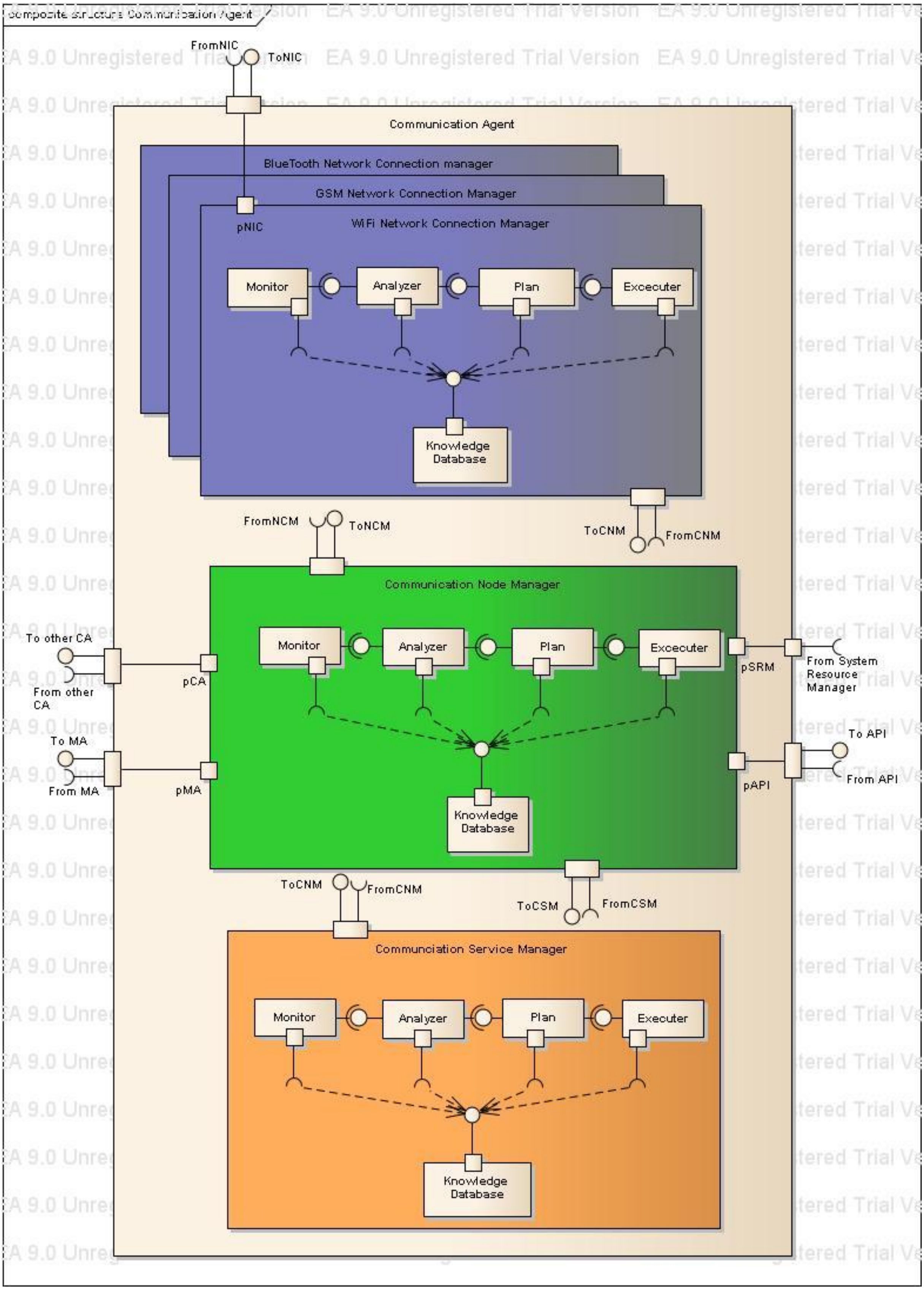

Figure 4: composite diagram of Communication Agent

Connection Manager

The connectivity manager's task is to create and maintain a channel for WiFi/GSM/BlueTooth .To transfer data, each channel uses a specific type of connection and when there is a change in the network context, the channel can be dynamically changed according to the local decision module.

4.2 MAPEK

Monitoring Module and Analyzer

End-to-end network information is monitored including connection information, network parameters (availability, bandwidth, loss). By explicit query, this module collects this information from underlying infrastructure. The analyzer scans the log files and it interacts with the knowledge database to detect failures and alert the plan.

Plan and Executor

Plan is used to identify the adaptive policies according to the boundaries of decision once the context information is gathered. The decisions are stored in knowledge database for future action by invoking the history. It communicates to other functional modules to negotiate and prioritize the actions and finally it executes the adaptation. SWRL [9] rules are used to define the adaptation policy by the application designer and used at Mission Agent, MA (establishment of cooperation flow between the investigators among the different groups) or at CA (energy level constraint). Elaborated work of SWRL rules are presented in our past work. Executor is responsible for adapting the decision by properly linking the services and also triggers MA/API in case if there is a need.

4.3 Collaborative deployment

Collaboration is viewed as a committed effort on the part of two or more actors to devise a new solution for a decision task. Communication Node Manager is responsible for sending and receiving messages to and from other devices, by abstracting over the specific routing protocols. Offered services are accessible to other devices and can be coordinated and composed cooperatively. Some of these services are applications that don't require human intervention. Others act as proxies for humans (for example, the service for instructing fireman to follow a participant is a simple GUI that alerts the user by displaying a pop-up window on his device or by emitting a signal).

Local measurements like resource usage monitoring (CPU & memory) are handled to estimate its own health status. At distributed mode, a device may cooperatively work with its neighbors to provide some measurements, such as accurate link quality. Device overhears its neighbor's transmission to collect relevant statistics like signal strength, dropped packets, etc. for problem diagnosis. At globally, it is necessary to collect measurements such as network topology and routing state, to detect routing anomalies. With sufficient measurement, it is probable to locate the problem and determine whether it is a link or device failure, or traffic congestion. Problem diagnosis determines the root causes from observed symptoms, such as abnormal events derived from the network measurements. The idea here is to use a knowledge database, which contains diagnostic rules to perform systematic analysis.

Coordinator's CA would direct a "bridge" device to follow the device/PDA that's going out of range, maintaining the connection and ensuring a path between the devices. In this way, the CA, on the basis of the disconnection prediction, schedules the execution of new, unforeseen activities. Such an adaptive change of the process is managed by the coordinator, which has knowledge about the status of all the devices and takes into account idle devices, operations that can be safely delayed, and so on. The issues addressed by collaborative deployment:

• at the Application/Technology level:

- Exchange and management of information.
- Adaptive management combines local connection management among devices.

• at the Architectural level:

- Devices communicate without relying on any infrastructure and sharing reliable communication channels in order to coordinate different teams.

4.4 Context management

Context adaptation is needed especially when mobile devices are collaborating through heterogeneous environment. Parameters like connectivity, user availability and location play a major role in order to anticipate provide an adaptable service. A decision is used to help extracting generalized rules from a variety of contextual information.

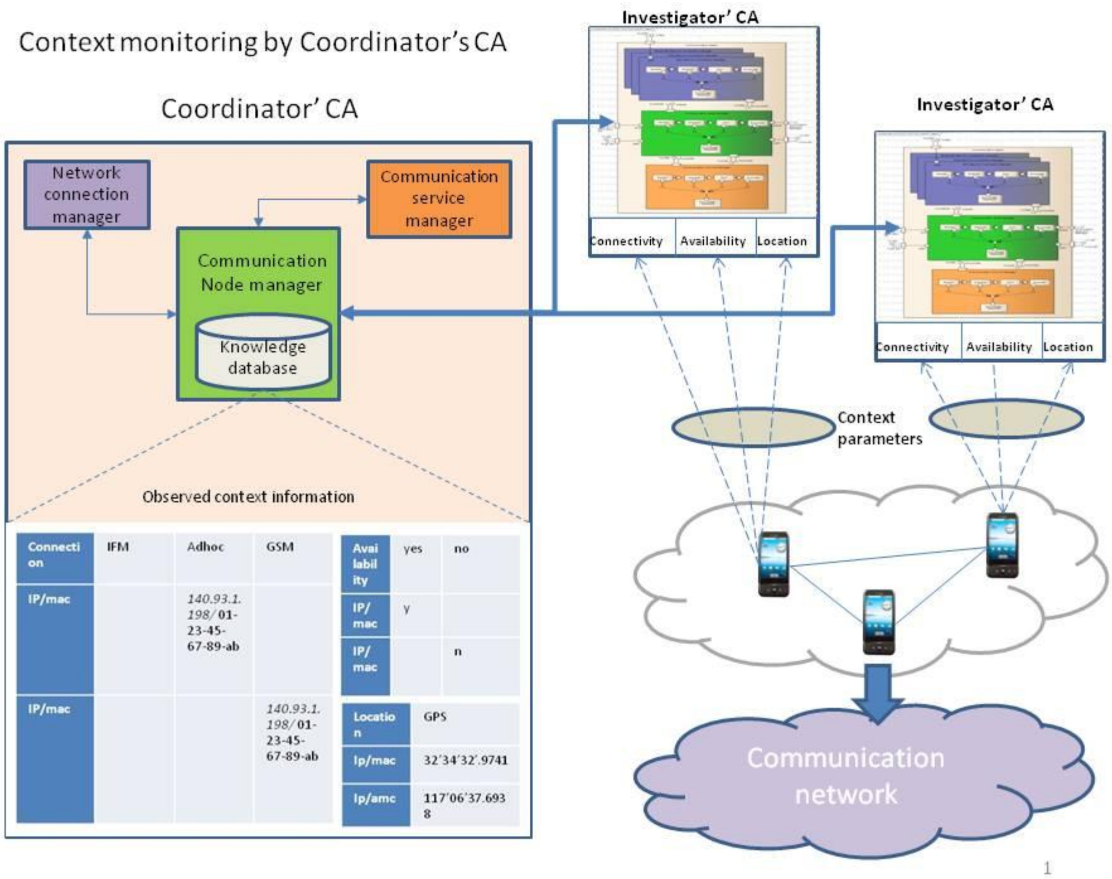

Figure 5: Context monitoring by Coordinator's CA

We define the context as a set of constraints, associated with the environment. These constraints should be considered so as to provide a better service or a more suited one. They can be related to the mobile terminal (memory), or to the interaction relations between users (connection mode, availability). One variation of the values of these variables (constituting a particular contextual situation) can significantly have influences on the expected behavior of the system. Figure 5 shows the coordinator's CA role in terms of context monitoring.

Supervisor and coordinator maintain a consistent state of the network and of each participant participate in the network. It manages the network topology (and its predicted next states) and the tasks each actor is in charge of, as well as services that offer. On the basis of that information, the coordinator applies algorithms for choosing a bridge and/or executes workflow task reassignment when needed. The coordinator's CA manages situations when a participant is going to disconnect, by applying algorithms for choosing a bridge, and by executing workflow schema restructuring and workflow task reassignment when needed.

5 Operational scenarios

One of the first tasks to be achieved on site is to deploy a local telecom network which provides the global network connectivity for actors involved in the mission. Access point is installed in a truck or robot which acts as a coordinator which propagates the network ID.   CA embedded in access point takes the responsibility of creating the local network automatically.

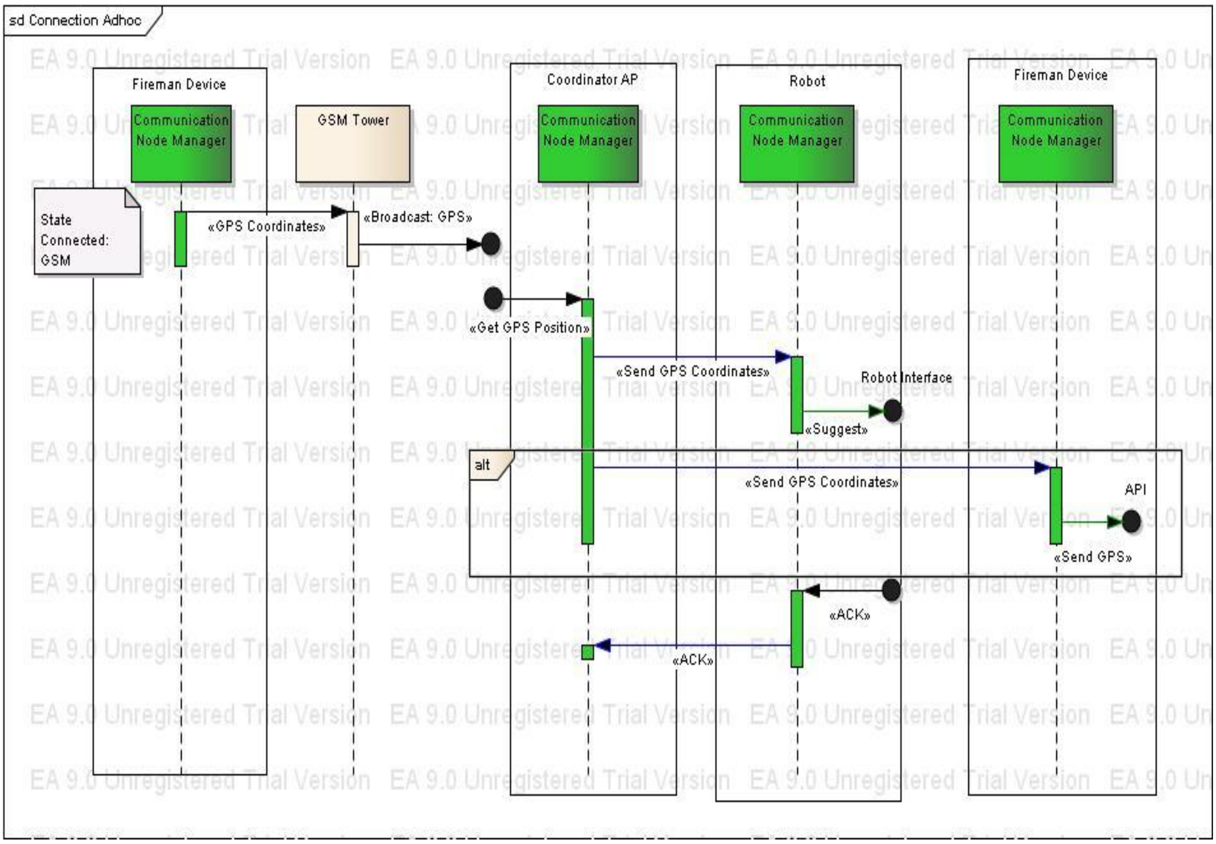

Figure 6 : Adhoc deployment sequence diagram

We deploy the WIFI network as the primary one and in case of failure of WIFI, CA chooses the available network access as secondary one (GSM). For example, sending video data using GSM is not recommendable as it lacks performance. Thus GSM acts as a substitution network means when there is a connection loss at primary one, GSM will be used to reactivate the WIFI network. Switching between WIFI status (IFM or Adhoc) depends purely on the status of the mission. Let's consider a lost fireman's CA sends its GPS position to its coordinator through GSM. The coordinator chooses one of the connected robots and send request to move to a position to get connected with the lost fireman. After receiving the message, robot's CA changes its IFM to Adhoc (Fig 6). Thus, when the robot approaches the Fireman's CA, the connection will be automatic as the device is already in Adhoc mode. The time dependant of this activity will be detailed in the plan.

If the device is in non-connected state, communicationNodeManager retrieves information from the NetworkConnectionManager about the last connection status. This information contains signal strength of available access technology. According to the policies, CommunicationNodeManager send message to NetworkConnectionManager to activate the second interface card. The decision could be local if the mode changes from IFM to Adhoc mode. The messages between communication between NetworkConnectionManager and CommunicationNodeManager should be periodic. Smart plan tries to take action if the module forsees a future connection loss. Trigger will be passed in between the modules when there is no connection after a threshold time out.

Improving team routing management by detecting lose of connection

The aim is to define the activities and actions of the CA for detecting connection lose, then finding out possible solutions and communicating these solutions to decision.

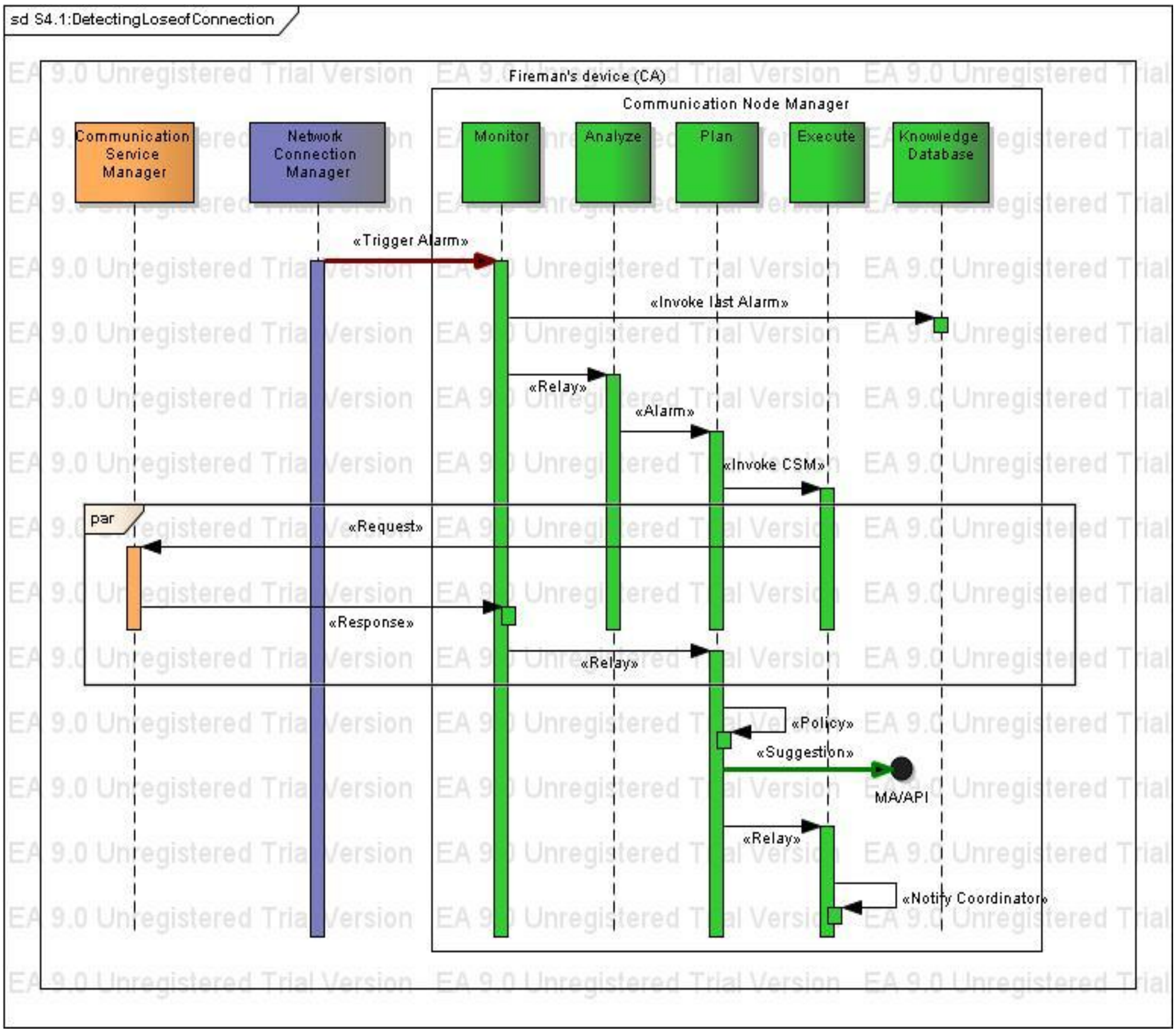

Figure 7: Sequence diagram for detecting connection loss

Hypothesis

- Signal history and its GPS position

ROSACE device possess some special application for its GPS coordinates. Thanks to this special application, CA invokes the XY coordinates and stores in its database. Whenever actor's device is connected to its peer or to its coordinator, communication node manager stores the XY position and its signal strength. Signal strength will be calculated by network connection manager. By storing the history of GPS position, CA can suggest the mission to take an action in case if it detects the deterioration in the signal strength. CA's plan should be strong enough to estimate locally the trajectory where the connection will be fine.

- Notify peer

CA can also notify its peer when there is connection deterioration before the loss or shift to adhoc mode to connect with its peer to calculate the good position.

Detecting connection loss

If there is deterioration in signal, there will be a trigger from network connection manager to communication node manager. Once it receives the trigger, node manager consults the database regarding the solution (thanks to the message stored last time when it was triggered) to find a solution. To confirm the loss, node manager consults with service manager if there is a loss in QoS. After the message communication, node manager suggests mission agent about the coordinates where it can have the maximum signal and then it notifies to coordinator (Fig.7).

5.1 Implementation

We use Vnet to simulate our scenario. It is the Virtual network being deployed to comprise all processes that simulate or implement the actual data transport layer, from message sending to delivery. This virtual network provides all necessary networking facilities to agents in the context of the ROSACE actions. This network is not intended for direct agent use, but to guarantee full monitoring and control of the experiment. The vNet then simulates any lossy network model during experiments. The existence of this network, however, makes possible the use of the same setup as in a centralized simulation, if desired. This implies logging and synchronization implementation (which can happen at a central place).

As a possible implementation, agents can use YARP to communicate with the single vProxy. YARP is in this case using TCP over the guaranteed backbone. The vProxy is then able to reliably simulate any kind of network models or failures needed for the experiment. In order to do so, the vProxy may need information about the robots (e.g. location). This acquisition is done through a particular instance of a vWorldModel which gathers this information from the robots.

Vproxy: An entity locally available to every agent, which provides the API necessary to make use of the vNet. All interactions between an agent and the vNet are done through a vProxy.

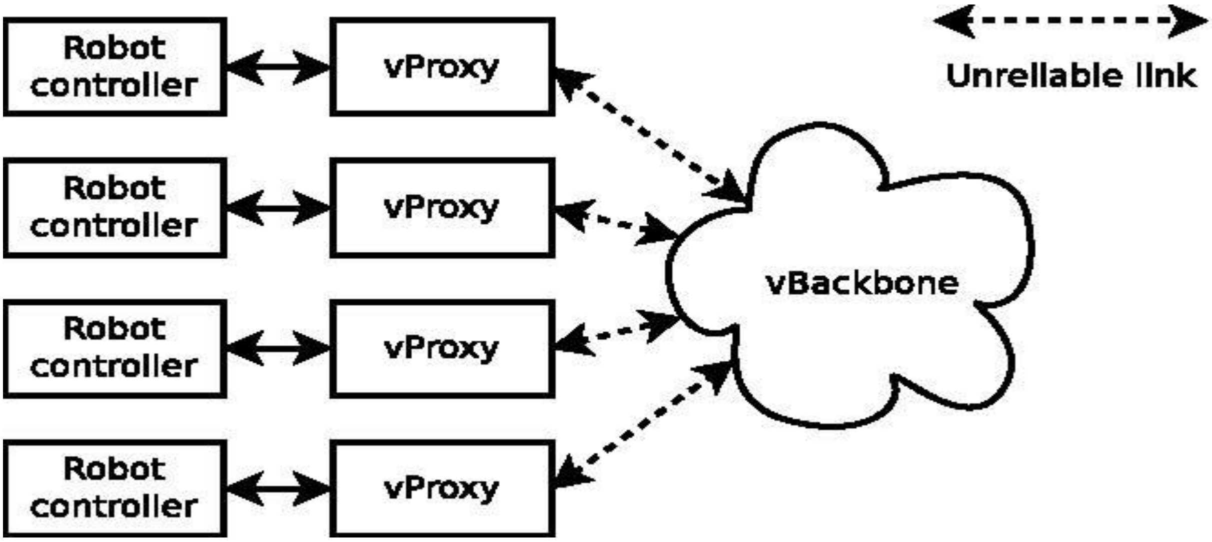

Figure 8: The deployment architecture of ROSACE devices

6 Conclusion and Open Issues

The adaptive communication in a collaborative environment is retained through the standard MAPEK architecture. Here we presented a telecom agent that does autonomic computing to ensure communications among ROSACE devices. The different modules in MAPEK are executed locally in investigator and distributed with peer entities. The agent among use appropriate policies to take an action executed without manual intervention. In case of environment change, triggers play an essential role to notify the decision components for initiating the adaptive policies. The implementation has some issues like dramatic change in the situation may not be known apriori. Other related problems include adding new investigators to a group dynamically cannot be predicted by a coordinator before.

Acknowledgments

This research is supported by the French project Rosace (RObots et Systémes Auto-adaptatifs Communiquants Embarqués).

References

[1] N. Nasser and H. Hassanein. Adaptive bandwidth framework for provisioning connection-level qos for next-generation wireless cellular networks. Canadian Journal of Electrical and Computer Engineering, 29(1):101–108, 2004.

[2] J. Z. Sun, J. Tenhunen, and J. Sauvola. Cme: a middleware architecture for network-aware adaptive applications. In Proc. 14th IEEE International Symposium on Personal, Indoor andMobile Radio Communications, volume 3, pages 839–843, Beijing, China, 2003.

[3] E. Exposito, P. Senac, and M. Diaz. FPTP: the XQoS aware and fully programmable transport protocol. In Proc. The 11th IEEE International Conference on Networks (ICON'2003), Sydney, Australia, 2003.

[4] L. R. Welch, M. W. Masters, L. A. Madden, D. T. Marlow, P. M. Irey, IV, P. V. Werme, and B. Shirazi. A distributed system reference architecture for adaptive qos and resource management. In Proceedings of the 11 IPPS/SPDP'99 Workshops Held in Conjunction with the 13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Processing, pages 1316–1326, London, UK, 1999. Springer-Verlag. ISBN 3-540-65831-9.

[5] D. Dhoutaut, "Etude du standard IEEE 802.11 dans le cadre des réseaux ad hoc : de la simulation à l'expérimentation", thèse, Institut National des Sciences Appliquées de Lyon, 2003.

[6] M. Ouzif, M. Erradi, H. Mountassir "Description of a Teleconferencing Floor Control Protocol and Its Implementation" Journal of Engineering Applications of Artificial Intelligence, EAAI, Editor, Elsevier, 2008.

[7] T Gu , H K Pung , D Q Zhang , X H Wang, A Middleware for Building Context-Aware Mobile Services, Vehicular Technology Conference, 2004, IEEE 59th.

[8] http://www.ginkgo-networks.com/IMG/pdf/AC_Blueprint_White_Paper_V7.pdf

[9] 12. Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., Dean, M.:SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission 21 May 2004 (2004)