Distributed Architecture for Applications based on the GSM Short Message Service

Giovanni Martini, Giorgio Rosenga
CSELT
Via G. Reiss Romoli 274 - 10148 Torino (Italy)
e-mail:{martini,rosenga}@cselt.stet.it

Abstract

This paper presents a distributed architecture for the definition and the deployment of applications based on the Short Message Service of the GSM network, and illustrates a first prototype realization using the OSF/DCE.

1. Introduction

Mobile communications are quickly changing the way people work and communicate. Increased is the need for new, personal and ubiquitous services.

The pan european standard for mobile communication, GSM [1], is expected to play a major role in the scene of personal communication systems. While the high expectations of the final users on mobile communications ask for new services, the urge for new infrastructure architectures is increasing in a parallel way. These new approaches come in help to network operators and to service providers in order to ease their task of definition, deployment and management of new services.

Bearing in mind this facts as the reference scene, and focusing on the GSM network in particular, we devised a distributed approach for a particular class of services, based on the GSM Short Message Service, or SMS, which is on the way of fulfilling the previous requirements.

This paper illustrates the usage of OSF/DCE [3,4] as the communication framework of a distributed architecture for the definition and the implementation of applications based on the GSM Short Message Service.

The paper is organized in the following way: after a brief overview of the GSM network and of the services it can support, we focus on the characteristics of the SMS, and on the possible set of applications which are based on. Furthermore, we compare the traditional approach in the definition and realization of applications with the distributed solution. A more detailed description of the solution with the DCE, is then illustrated.

2. GSM: the network and the services

The GSM network is the pan european standard for the mobile communications. In the next sections its main characteristics are illustrated.

2.1 The GSM network

The GSM system has been the first standardized system that uses digital transmission for the radio channel. This makes it compatible with the digital evolution of the fixed networks.

Without deepening too much into details, the main elements of a GSM network are illustrated in Fig. 1. They are:

- the mobile station (MS): it is the mobile terminal used by the final user to access the GSM network and to make use of its services;
- the *base station system* (BSS): it is the network element which converts the radio signals from the mobile terminals into signals for the fixed network; it provides for the management of the radio resources over a single cell, and for the switching between the radio channels and the connections with the mobile switching centers;
- the *mobile switching center* (MSC): it is the element which routes and retransmits the data all through the network; it is connected to the local public switched telephony network (PSTN) to provide the connectivity between the mobile and the fixed telephony users.

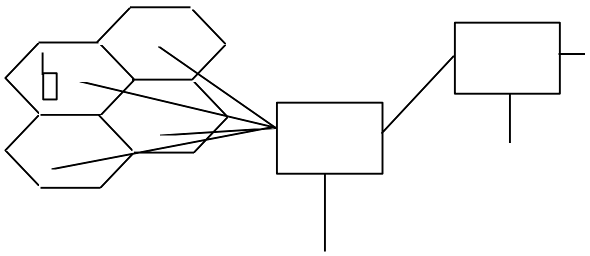

Fig. 1: GSM cellular infrastructure

The flow of data in the network uses two different logic channels:

- the traffic channel, for the transport of the coded voice signals and for the user data;
- the signalling channel for the transport of control and synchronization information between the base station and the mobile terminal.

Another important point regarding the data transmission is that each packet transmitted over the network is encrypted, in order to achieve reliable security levels.

2.2 The Short Message Service

The GSM network offers to the users some base services in addition to the bare voice communication. Among them, the most important is the Short Message Service. It allows the transfer of textual messages (with a maximum of 160 characters) or binary data (with a maximum of 140 bytes) between two mobile systems, one of which has to be a user of the GSM network.

The most considerable characteristics of this service are:

- the data transmissions do not interfere with other activities (e.g. conversation) because they use the signalling channels;
- the encryption functionalities may be used in order to achieve security constraints.

All the short messages are transmitted through the Short Message Service Center [2] (SMS-C or Service Center). Its purpose is twofold: one of its functionalities is the provision of interworking capabilities between the mobile network and the local public fixed network. Furthermore, the Service Center offers a store and forward mechanism for each message: if the mobile station destination of a message is off-line, the Service Center retains the message until the mobile station becomes again active, or the timeout indicated in the message expires.

The SMS, according to the ETSI GSM recommendations (TS 03.40 and TS 04.08), is a point to point service specialized in the two following types:

- mobile originated, for message transfer from the mobile station to the Service Center;
- mobile terminated, for message transfer from the Service Center to the mobile station.

At the moment, only mobile terminated messages can be sent over the network, because ETSI operational specifications are in the Phase 1; complete functionalities is envisaged in the Phase 2, the second and final stage of the specifications.

The short messages may be simple data or commands for the Service Centers. In the first case, they transfer information between mobile stations and different applications; whereas in the second one, the commands let the user interact with the Service Center itself (e.g. query of the status of a previously submitted message).

The main elements which characterize each message are:

- *validity period*: it is the information regarding the maximum time the Service Center must try to send a message to a given destination in case it is not readily available (e.g. the mobile station is momentarily disconnected);
- *protocol identifier*: it permits to identify the type of application to which the Service Center sends the message (e.g. fax transmission);
- *user data*: the message to be transmitted;
- *origin/destination address*: the addresses of the sender and of the receiver.

In [2] the results on SMS transmission delays and throughput are illustrated. According to the reported measurements, the average delay for a message from a mobile station to the SMS-C is around three seconds. This value leads to a low throughput rate, which characterizes the SMS as a service suitable for the transfer of small amount of data.

2.3 Services based on the Short Message Service

The SMS may be used as the basic service for building other different applications for the data transmissions. Though it was conceived as an advanced paging service, it may originate a wide selection of valued added services, all being characterized by small amount of data exchanges, needing at different degrees some levels of security on the transmitted information, and leaving the normal traffic channels available for other purposes.

One class of applications regards TeleAlarms and TeleMeasurements: it is possible to envisage a system where sensors may signal state variations sending information via the Short Message Service and the GSM remote terminals associated with the sensors. The range of applications may include, for example, the intrusion control, or fire control.

Another application class is support for the transmission of banking transactions, directed to banks or shops with strong requirements on mobility, rapidity and security, all of them fulfilled by the short message carrier.

One of the applications we have focused on in our prototyping, is the electronic mail, one of the most common service available on the fixed network. The GSM is emerging as the global network interconnecting the users of the fixed and of the mobile networks. It is natural to envisage the electronic mail as one of the first applications which is going to extend the services based on the short message.

3. Software architecture of the Service Center

In the next paragraphs two different application architectures will be compared. The traditional one, which derives in a natural way from the more general view of the GSM framework, and one more innovative, which distributes and widens the range of the applications of the SMS-C. Following the comparison of the two approach, a more detailed description of the distributed one is illustrated.

3.1 The traditional architecture

The traditional architecture of a SMS-C , with some application installed in it, is depicted in the Fig. 2.

For the sake of simplicity, it is supposed that the user nodes and the Service Center are connected only by the TCP/IP protocol, while the SS7 protocol is used to communicate between the Service Center and the mobile switching center (of course other communication protocols may be used to connect the Service Center with the fixed network) .

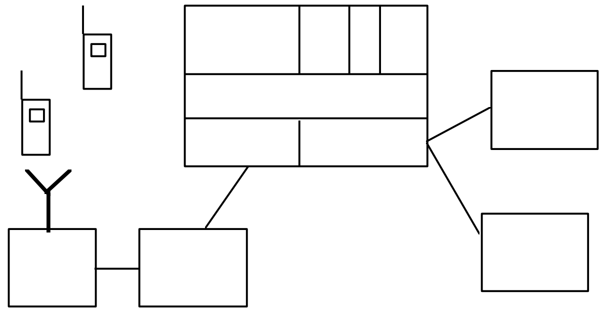

Fig. 2: monolithic architecture of the Service Center

The traditional application architecture here illustrated, strongly monolithic, is composed of the following main modules:

- *message kernel*: it is the core of the SMS-C and implements the store and forward functionalities;
- *management*: the management functionalities for the control of the SMS-C itself and of the installed applications (e.g. deploying a new version for a given application, or changing some runtime parameters of the message kernel);
- *application*: it is the set of the applications, each one identified by its own protocol identifier;
- *user*: it is the front-end program that interacts with the final user and the application on the SMS-C.

This solution requires that all the applications are installed in the same node of the SMS-C. This fact imposes some constraints on the global application architecture, which may be summarized in the following points:

- limit on the usage and availability of the resources in the SMS-C node for each installed application, which may lead eventually to some constraints on the overall performance;
- lack of openness of the configuration: in this way the applications are strongly integrated with the SMS-C and with the interface of the message kernel made available by its manufacturer. This fact makes really hard trying to integrate applications available by third party software providers. This point represents a significant issue for the network operators that do not want to be tied to a single manufacturer.

3.2 The distributed architecture

The proposed distributed architecture is intended to give a solution to the inconvenients of the traditional one.

In Fig. 3 a schema of the distributed solution is depicted. The main idea is to decouple the principal components of the SMS-C and to remote them on different hosts, maintaining at the same time the possibility for some application to be directly installed on the SMS-C itself.

The main advantages of such a solution are:

- openness of the SMS-C: having made public the interface of the message kernel (not only from a specification point of view but also from the interoperability one), it is now possible a greater flexibility in the introduction and in the development of new applications, not being anymore constrained only to the ones supplied by the SMS-C manufacturer;
- increased fault tolerance: it is possible to have redundant copies of the same application on different hosts, and henceforth to gain an increased reliability of the whole system;
- platform independence: the platform transparency is realized by the middleware used for the connection;
- increased parallelism: applications located on different nodes do not interfere one each other, contributing to a final increased level of parallelism;
- application sharing: on a GSM network may be present multiple Service Centers, each one serving a given area; in the case of particularly critical applications (e.g. a nationwide earthquake alarm system) the same application may be shared among all, or part of, the Service Centers, instead of having multiple replicas installed in each Service Center.

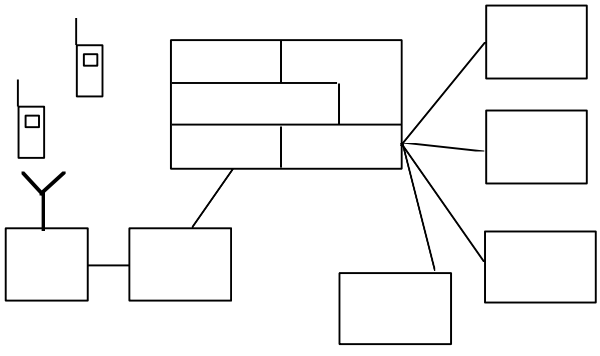

Fig. 3: distributed architecture of the Service Center

With respect to the previous benefits, the costs to be paid are the usual ones for distributed solutions, and in particular:

- data security: reserved data may be contained in the short messages (e.g. information regarding a banking transaction), and with the distributed approach the possibility of fraudulent interceptions increases;
- reduced performance due to the communication delays between the Service Center and the hosts on which the applications run;
- problems of application management: all the software in the different hosts must be kept consistent (also in case of different software providers);

The solution described here is independent from a specific technology adopted for the support of the distribution.

In the following section we illustrate in greater detail the solution with the DCE platform. In the near future other interesting solutions could be based on different technologies, among which the one based on OMG/CORBA [5] reveals interesting aspects.

4. Solution with the OSF/DCE

The solution based on the DCE is illustrated in the following schema. It is straightforth derived from the previous one. As explained in the previous section, DCE is used to support the distribution of the main entities of the SMS-C, and it is not intended as a porting of the DCE over the SMS.

The main components envolved in this solution are:

- *agent process*: it plays a gateway role, sending the short messages read from the message kernel to the specific applications they are for (and viceversa), while notifying the accounting process of the transmission of each short message;
- *application process*: the applications may be either local to the message kernel (as it happens in the traditional solution) or distributed on different hosts;
- *accounting process*: it collects from the agent the information regarding each short message sent, in order to produce off-line billing information.

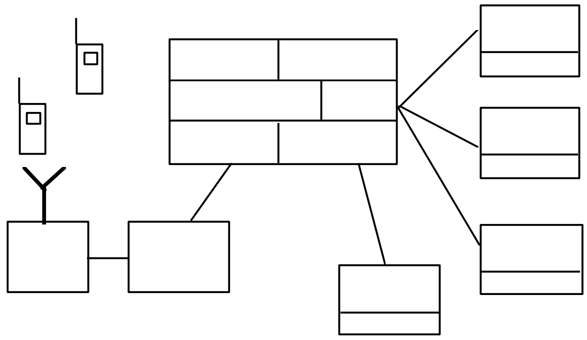

Fig. 4: distributed architecture with DCE

As previously stated, the distributed solution is independent from the adopted technology for the integration of the applications and the agents. Though the choice of a particular type of platform (e.g. CORBA) does not affect much the basic idea of separating the applications from the message kernel, we decided to use the DCE mainly for the following reasons:

- hardware platforms independence and interoperability: the DCE remote procedure call (rpc) mechanism allows data exchanges between applications in a transparent way from the hardware they run on. The data passing is done automatically by the DCE platform in a transparent way. It is only required to define at high level, in an appropriate definition language, the interface between a client and a server.
- easy scalability of the configuration: by using the DCE it is easy to reconfigure the nodes on which the applications run. The nodes may be grouped in cells, which are, according to the DCE terminology, the basic units for the management and the configuration of the system. The size of a cell may be changed (i.e. nodes may be added or removed) at any time without affecting significantly the performances of the communication among client and server applications.
- reconfigurability: DCE provides the mechanism of directory service (cell directory service), which allows server applications to dynamically advertise their own interfaces in order to allow clients to retrieve them in a later time. In this way it is feasible relocating the applications within a cell, by using this characteristics of dynamic binding. Moreover, the relocation activity may be used as the basic mechanism for ensuring a higher level of fault tolerance .
- access security: for each application using DCE it is possible setting up access rights in order to avoid that a fraudulent application may connect to an agent. In this way, it is possible the activation of the data encryption mechanism at the remote procedure call level in order to achieve the same security level on the fixed network as the one provided by the GSM network.

4.1 Details of the prototype realization

The prototype of the architecture we have devised focuses on the possibility of a n easy and practical configuration of the services deployed on the network. Main emphasis has been put on the fact of being able to relocate applications from one node to another without having interruptions in the service delivery. Nevertheless, the issue providing a complete and general fault handling mechanism is far from being solved. For the moment, only the basic mechanism is provided, but the policy and the supervision activities needed to provide a stable and reliable fault handling, require further investigation.

In order to meet the requirements for a robust configuration, we decided to use the DCE directory service as the mechanisms for allowing dynamic association between agents on the SMS-C and applications on the network.

Moreover, by using the access control list of the directory service it is possible to ensure that only authenticated applications may interact with the agent process on the Service Centers.

Some further explanation is needed on the mapping between the interfaces supported by the applications and of the agents and the protocol identifiers used to identify single applications. In the DCE directory service the following entities are stored: the interface uuid (unique universal identifier, provided by the DCE), the object uuid and the interface reference (i.e. where the server with the specified interface is located in the network). The search mechanism uses the interface uuid and the object uuid as keys during the retrieval process. In order to realize a general mechanism enough flexible to be used with different services, the following mappings have been adopted:

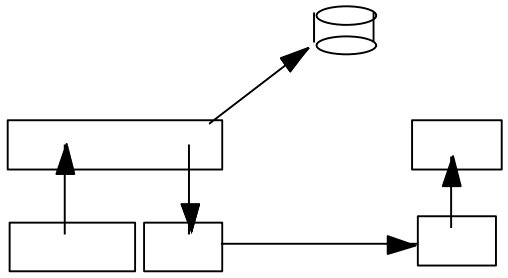

Fig. 5: communication between an agent on the Service Center and a remote application

• there are only two kinds of DCE interfaces, one provided by the agent process and one by each single applications: each one is distinguished by its own uuid;

• a mapping exists between each protocol identifier and an object uuid: this mapping is static, and is decided once for all during the first installation of a new service.

In such a way, a server (agent or application) knowing its protocol identifier, is able to compose the key (interface uuid - object uuid) by which searching in the directory service for the address of the correct counterpart with which is going to communicate.

In Fig. 5 the main phases of a sample communication between an agent and the correspondent application are illustrated. The reciprocal stands for the communication in the opposite direction. The phases are:

• 1) the agent process periodically polls the message kernel for new short messages: in case of their presence, it picks them up one at a time (the polling of the message kernel is due to its actual prototypal implementation used in our experiments: next releases will make use of asynchronous reads of the short messages);

• 2) after having read a short message from the message kernel, the agent process tries to send the message to the application by using the DCE rpc mechanism (it has been supposed that the first import from the directory service of the counterpart's address has already been done during the startup phase); in case of failure (e.g. the application is not running anymore, or a communication failure has occurred), the agent performs the next phase 3, otherwise the execution continues to phase 4;

• 3) the agent imports again a reference (the same as before, or possibly a new one if a backup application replica is running) from the directory service for the class of service it supports;

• 4) the rpc mechanism transfers the data;

• 5) on the application side, after having received the data from the DCE runtime, a new thread is activated, and it performs the proper code in order to exploit the service.

4.2 A test application

In order to validate the proposed architecture, a first prototypal application has been realized.

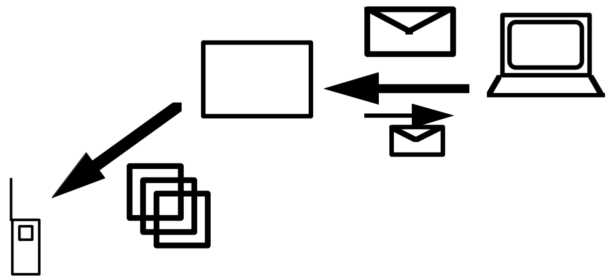

Fig. 6: email service using the Short Message Service

The scenario of the application is the following one. The goal is being able to send emails from any fixed terminals connected to the fixed network to GSM users, who have subscribed for this service. Each one of them is identified by the number of his/her cellular terminal (e.g. the user having a terminal with the telephone number `12345` is recognized by the email system as `12345@service_center.cselt.com`).

Due to the reduced size of the data transmitted by a single short message, each mail received by the email application is splitted into one or more short messages. Each one is sent by the Service Center to the GSM user. The mails that are too large to be splitted are not sent over the network: only a notification message indicating the subject of the mail is sent to the user. The limit on the size of the mails that can be decomposed in short messages is imposed by the low throughput rates of the SMS.

For each mail sent to a GSM user, the sender receives an acknowledge mail reporting the status of the transmission (e.g. mail discarded because too large, mail successfully delivered).

5. Conclusions

In this paper we have presented a distributed architecture which provides basically openness and easiness of reconfiguration of the services based on the GSM Short Message Service. The proposal goes beyond the bare traditional architecture, allowing a more flexible and open structure for the deployment and the management of new services.

The proposed architecture has been realized in a real prototype which is based on the DCE distributed platform. The work is still under progress, but the first results and the applications realized to validate the whole architecture, have proved the feasibility of the distributed approach.

Though some work still needs to be done, especially from the administrative and management point of view, the results obtained by now encourage us in pursuing this solution.

Bibliography

[1] ETSI, "GSM Technical Specifications"

[2] S. Collesei, P. Di Tria, G. Morena, "Short Message Service based application in the GSM network" in IEEE/ICCC, 10.3, 1994

[3] Open Software Foundation, "OSF DCE - Application Development Guide" , Cambridge, Prentice Hall, 1993

[4] Open Software Foundation, "OSF DCE - Application Development Reference" , Cambridge, Prentice Hall, 1993

[5] Object Management Group, "The common object request broker: architecture and specifications" , OMG Doc. No. 91.8.1, august 1991