Radio Access Network Traffic Generation for Mobile Switching Center

Suliman Albasheir and Sofiène Tahar
Department of Electrical and Computer Engineering
Concordia University, Montreal, Quebec, Canada
Email: {s_albash, tahar}@encs.concordia.ca

Claude Gauthier and Jean Roussel Personna
Ericsson Research Canada, Montreal, Quebec, Canada
Email: {claude.gauthier, jean.roussel.personna}@ericsson.com

Abstract— One of the challenges faced by telecom companies is to provide robust and powerful servers that are capable to handle the great increase of the number of subscribers and to accomplish the heavy Internet-based applications that generate a tremendous traffic load. Companies evaluate their products' performance before releasing them to the market by applying a large amount of *generated traffic* to the telecom servers in order to measure their capability under traffic load; powerful solutions are hence needed for generating traffic and modeling various telecom protocols. In this paper, we propose a new traffic generator solution to load the Mobile Switching Center (MSC) for the Universal Mobile Telecommunications System (UMTS). This traffic generator loads the MSC through various mobile call scenarios such as location update, mobile call originating, mobile call terminating, and call clearing. We utilize the UML Use Case Model to describe the functional behaviors of the traffic generator, and present the UML Analysis Model that provides the logical implementation of the functional behaviors of the proposed traffic generator.

Index Terms—Mobile Network Traffic Generation, Mobile Switching Center, Load Modeling and Testing, UMTS Call Scenarios, UML.

I. INTRODUCTION

The telecommunication industry has been growing tremendously during the last decades. The International Telecommunication Union (ITU) reported that worldwide mobile cellular subscribers have reached the 4 billion mark by the end of 2008 with an estimated mobile penetration rate reaching 61%. Also, it is reported in [12] that the global telecommunication revenue have reached $2.0 trillion USD by the end of 2008, an increase of 7.6% over revenues in 2007. On the other hand, with the new next generation networking technology, more and more Internet-protocol based applications are introduced or planed to be implemented in the future.

Consequently, all these parameters require powerful telecom servers that are capable to handle the tremendous increase of the subscribers' number and to carry out the heavy next generation applications that generate huge traffic loads. It is extremely important for telecom companies to have powerful servers that are capable to carry out the huge telecom traffic loads. To achieve this, testing engineers in telecom companies apply huge amounts of simulated traffic to their telecom devices to measure their capability and behavior under load pressure; this kind of test is called *Load Testing*. Based on the load testing measurements, design engineers can modify the design of the system to have more robust systems under the traffic load. To conduct load testing, powerful solutions are required that simulate telecom protocols, initiate call scenarios, interact with the telecom network, as well as generate traffic messages towards the corresponding device under test. This operation is called *Traffic Generation*.

In this paper, we introduce a new traffic generator solution to exercise load testing for the Mobile Switching Center (MSC) and measure the capability and performance of the MSC under traffic pressure. The MSC server is one of Ericsson's WCDMA/GSM Core Network components that controls the setup and the release of the communication connections in the network. The proposed traffic generator provides models for the Radio Access Network Application Part [3], the Mobility Management, and the Call Control protocols [4] for the *Circuit Switched* network.

Figure 1 shows the traffic generation environment for loading the MSC. For such traffic load, many approaches model the whole Universal Mobile Telecommunications System Terrestrial Radio Access Network (UTRAN) components' protocols [5]. These components are: the User Equipment (UE), the radio base transceiver station (Node-B), and the Radio Network Controller (RNC). Obviously, it is not cost effective to simulate all these components and their protocols in order to load the MSC server.
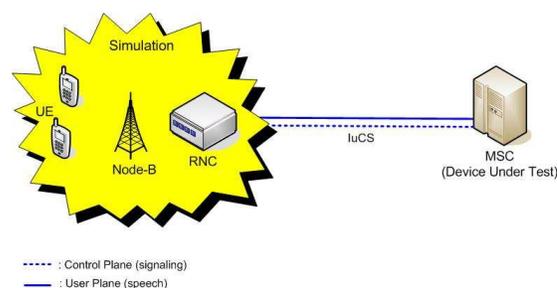

Fig. 1. Traffic Generation Environment for the Mobile Switching Center

To simulate an environment around the MSC for load testing first, we have to simulate the Media Gateway (MGw) that handles the Gateway Control Protocol (GCP). The MGw Simulator (MGwSim) is a tool developed by Ericsson that interacts with the MSC and generates GCP traffic [9]. Second, we have to generate the messaging traffic which simulates the RNC and other components of the UTRAN (Node-B and UE). To generate this traffic, testing engineers at Ericsson use either MGTS [14] or 3GSim [10] traffic generation tools along with the MGwSim to generate messaging traffic that properly loads the MSC server.

Our proposed traffic generation technique simulates only the messages on the interface between the Radio Network Controller (RNC) and the MSC server; the interface between the RNC and the MSC, called IuCS, is used for circuit switched data transfer. Furthermore, our proposed design for traffic generation is modeled based on an Ericsson proprietary platform called Connectivity Packet Platform (CPP) [8]. CPP is a very powerful node used for emulation and protocol transportation in many applications within Ericsson globally, and provides a high rate of calls generation. It is expected that our proposed design economizes in using real telecom equipments; since our design provides models for the radio access network messages, so it does not require real components such as the RNC to generate the traffic.

Several languages and techniques have been used to model telecommunication standards protocols, such as Specification and Description Language (SDL) [6], Message Sequence Charts (MSC) [7], or Unified Modeling Language (UML) [2]. In this paper, we make use of UML to model the traffic generator protocols and call scenarios. Furthermore, we use the Rational Systems Developer (RSD) from IBM [1] to build the UML Use Case Model and the UML Analysis Model.

The rest of the paper is organized as follows. Section II reviews some related work. Section III describes details of the methodology we follow for modeling the traffic generator. Section IV presents the proposed architecture and the main components of the traffic generator. Section V describes the UML Use Case and Analysis Models of the traffic generator. The last section concludes the paper and provides hints for future work directions.

II. RELATED WORK

There exists a number of related work in the area of protocol modeling and traffic generation for telecommunication networks using various techniques and methodologies. For instance, the work in [17] proposes a model for a traffic generation tool that has been used to evaluate the performance of some applications between GPRS and WLAN networks. This work generates traffic by having traffic profiles produced by user-level software modules. This work is an extension of a previous work described in [18] that presents an architecture for a traffic generator capable of generating Asynchronous Transfer Mode (ATM) traffic according to stochastic models. In the work presented in this paper, we propose a design for traffic generator by modeling the RANAP, Mobility Management, and Call Control protocols without using traffic profiles information.

In [19], the authors describe a software that simulates mobile call scenarios such as mobile originated calls and mobile terminated calls for UMTS. The provided simulator is capable to generate RANAP messages on the interface between the MSC and the RNC. Using the OMNeT++ simulator [20], some of the UMTS components have been simulated; such as mobile equipment, Node-B, RNC, Media Gateway, MSC Server, and others. Also, this work describes the functionality of the simulated communication system using SDL. However, this software still cannot be used for testing real equipment as it would require connectivity functionalities to the real UMTS communication servers, for instance the MSC server. Our model is capable to interact with the MSC server, since it is built based on the CPP which provides a connectivity functionality with the MSC.

In [21], the authors propose a modeling technique for location operations in a UMTS Network. These operations belong to both core and access networks under the Mobility Management protocol. For location operations modeling, the work of [22] provides functional blocks that represent various components in the UMTS, such as UE, Node-B, RNC, MSC, and others to simulate the location operations. The provided model still can be improved by implementing the designed test outputs in the presence of physical implementations of network elements. In contrast, our models are capable to communicate with real MSC through the CPP.

A few companies also provide commercial tools for traffic generation, which can be used for telecommunication systems load testing and performance evaluation. Some of these tools are MGTS [14] and DCT2000 [15] from Catapult Communications as well as Solver from Polystar [16]. These tools are computationally expensive and, unlike our proposed approach, they load the MSC based on modeling the whole UTRAN. Some other tools are 3Gsim [10] and UTMS [11] from Ericsson, which, however, require real components, such as RNC and MGw to load the MSC; this also makes the using of these tools to load the MSC costly.

III. METHODOLOGY

In real applications, many call scenarios load the MSC server such as; mobile location update, mobile call originating, mobile call terminating, call waiting and forwarding, roaming call scenario, and others; but the mobile originating call, mobile terminating call, and location update call scenarios can be responsible for most of the messages that load the MSC. Based on this, we consider those three call scenarios to generate the load traffic.

In the proposed traffic generator design, we concentrate on the modeling of the control plane messaging (signaling); no models will be provided for the user plane (speech). An external tool for user plane traffic generation will be used to provide the speech load. In this paper, we propose a model for the *RANAP Simulator Traffic Generator*, hereafter referred to as *RanapSim*. Figure 2 shows the UMTS network architecture and highlights the IuCS Interface that connects the RNC and the MSC. The transferred message over this interface contains many protocols' headers.
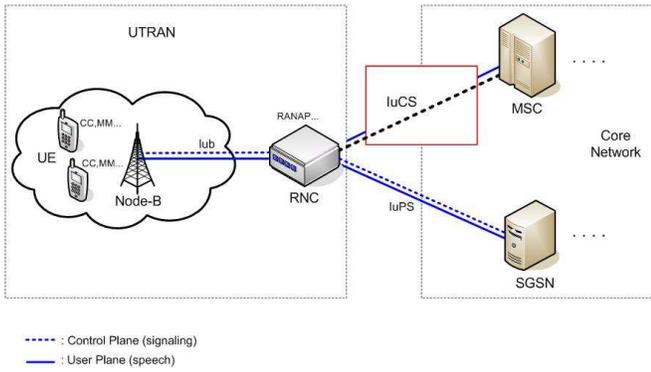
Fig. 2. The UMTS Network Architecture

Our methodology does not require complete models for the UE, the Node-B, or the RNC; since we model only the Mobility Management (MM) and the Call Control (CC) protocols' messages [4] (generated by the UE) and model the RANAP protocol's messages [3] (generated by the RNC), which makes our approach cost effective and more efficient. The other protocols (see Figure 3) do not need to be modeled, since we are taking advantage of using the CPP. This latter is used for execution and transport purposes; the execution part provides a support for software application execution, and the transport part provides connectivity functionality for several communication protocols [8]. It provides connectivity functions for the physical layer protocol, the ATM protocol, up to the SCCP (Signalling Connection Control Part) protocol.
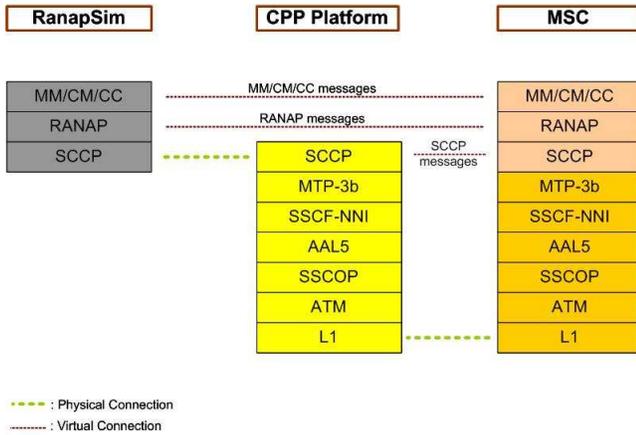

Fig. 3. Protocol Stack between RanapSim and MSC using CPP

Figure 3 shows the protocol stack provided by the CPP, the modeled protocols for the RanapSim, and the MSC protocol stack. This figure illustrates that the RanapSim provides models for MM and CC protocols which are responsible for representing the UE signalling messaging. These protocols should communicate with their corresponding protocols at the MSC server.

The RANAP protocol in the RanapSim represents the RNC signalling messages where the UE messages portions are carried as well. From Figure 3, it is obvious that RanapSim communicates with the CPP through the SCCP protocol that controls the signalling connections, where CPP provides an

Application Programming Interface (API) based on the UML to facilitate the communication with it [8]. By modeling an interface through the SCCP protocol, our model is able to communicate with and route messages to the CPP.

By handling the RANAP, MM, and CC messages to the CPP through the SCCP interface, the CPP is responsible for building the rest of the message by providing the headers of the SCCP, MTP-3b, SSCF-NNI, AAL5, SSCOP, ATM, and the physical layer protocols [5]; the whole message will be forwarded by the CPP through the physical connection to the MSC server. Once the MSC receives the message from the CPP, it will decode and process the whole message contents including the generated RanapSim message portion. So, by utilizing the CPP transport functionality, we are exempted to model the rest of the protocol stack and this makes our model more efficient. Using this technique, we are able to communicate and load the MSC by the RANAP, the MM, and the CC messages without modeling the whole UTRAN.

IV. PROPOSED ARCHITECTURE

Our RanapSim model should provide three main functionalities, namely:
- Generate signalling traffic (RANAP, MM, and CC modeling).
- Interact and interface with the CPP (SCCP interfacing).
- Control the interacting components.

Figure 4 shows the RanapSim main components which are the *Traffic Handler*, *RANAP Controller*, and *SCCP Interface Controller*, also it illustrates how these component are connected to the MSC server through the CPP; the RanapSim Manager is connected to the RANAP Controller component to manage all components' operations.

A. Traffic Handler

The Traffic Handler is responsible for generating the signaling messages for some call scenarios, such as location update, mobile originating call, mobile terminating call, and call clearing. To generate these signaling messages, the Traffic Handler provides models for the RANAP, the MM, and the CC protocols. These models build the contents of each message for the call scenarios that load the MSC server. This component is responsible for responding to any signaling messages received from the MSC through the RANAP Controller and sending the response back.

The main responsibilities of the Traffic Handler can be summaries as follows:
- Handle the *location update*, *mobile call originating*, and *call disconnect* call scenario requests and generate suggested connection identifier and connection state. The generated connection ID along with the connection state will be forwarded with all messages related to the generated call scenario. This ID will become the SCCP connection ID -at the CPP- which helps distinguish between different messages that belong to different calls.
- Provide models for the RANAP protocol which represent the Radio Access Network and for the MM and the CC
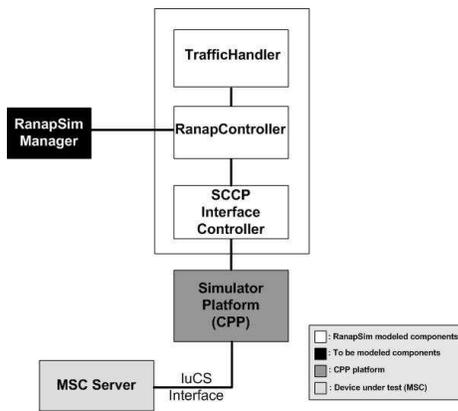
Fig. 4. RanapSim Main Components

protocols that represent the mobile radio interface. Those models are used to build the supported call scenarios and generate the required messages.

- Identify the received messages from the MSC by using the SCCP connection ID which corresponds to a specific call scenario. If the SCCP connection ID is unknown or not attached in the message, the Handler will look deeper into the message contents to understand the *message type* to perform the proper action [23].

B. RANAP Controller

The RANAP Controller represents the main controller of the system, since it controls all interactions between the traffic generator components and other external components. Also, it represents the connecting point between the RanapSim Manager, the Traffic Handler, and the SCCP interface Controller. The RANAP Controller's responsibilities are:

- Handle the *SCCP Attach* and the *SCCP Detach* requests. These requests are initiated by the Manager to attach/detach the SCCP Access Point Facade Interface (SCCPApfi), this interface handles the SCCP protocol's control plane (signalling) in the CPP.
- Handle the Manager call scenarios requests for the call scenarios by forwarding them to the Traffic Handler component in order to generate calls.
- Receive call scenario messages from the Traffic Handler and forward them to the Manager (for tracing and following up purposes). Also, the RANAP Controller forwards these messages to the CPP.
- Receive the MSC messages (responses/requests) which are sent through the CPP, forward them to the Manager (for tracing and following up purposes), as well as forward these messages to the Traffic Handler to complete the call scenarios.

C. SCCP Interface Controller

The SCCP Interface Controller is a component that deals with the CPP to control the SCCP connections, exchange messages, and interact with the CPP interfaces which are based on the SCCP protocol. This controller is responsible for

handling all messages from/to the CPP, and the CPP itself will take the responsibility to interact with the MSC to accomplish our target of applying load to the MSC.

The CPP provides various protocols that we can interact with, and for each protocol there are many interfaces to deal with. In our case, we are dealing with the SCCP protocol. The CPP provides two interfaces for the SCCP; namely, the control plane interface (SCCPApfi) and the user plane interface (SCCI) [8]. The SCCP Interface Controller's responsibilities can be summarized as follows:

- Attach the SCCPApfi interface in order to use the SCCP service. This attach request will be initiated by the Manager.
- Handle the SCCP connections for several call scenarios.
- Transfer data to the CPP in order to communicate with the MSC.
- Detach the SCCPApfi interface. No more SCCP services will be provided after detaching this interface.

V. RANAPSIM UML MODELING

A. UML Use Case Model

The UML Use Case Model describes a system's functional requirements in terms of Use Cases. It consists of all Actors of the system and various Use Cases by which the Actor interacts with the system. Each Use Case describes the functionality to be built in the proposed system, which can *include* another Use Case's functionality or *extend* another Use Case with its own behavior [2].

In this section, we make use of the Use Case Model to interpret the functional description of the *RanapSim Traffic Generator* into UML modeling elements. The Use Case Model considers the RanapSim system as a black box; this means that the building blocks within the RanapSim should not be mentioned in the Use Case Model.
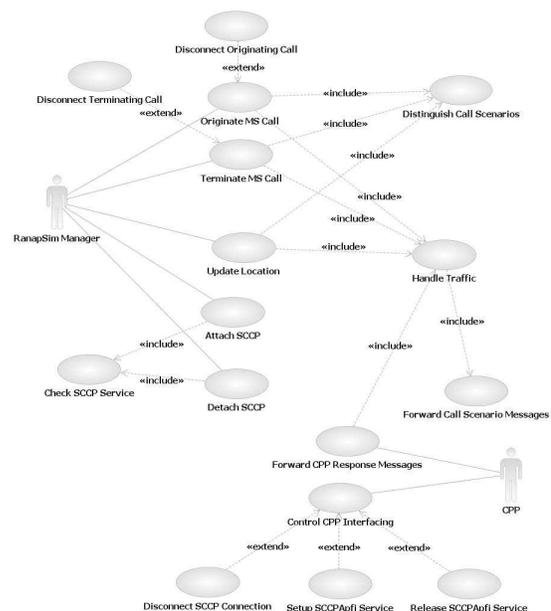

Fig. 5. RanapSim Use Case Diagram

Figure 5 depicts the Use Case diagram for the *RanapSim Traffic Generator* Use Case Model. It shows that the Traffic Handler functionalities are interpreted into seven Use Cases which are: the *Update Location*, *Originate MS Call*, *Terminate MS Call*, *Disconnect Originating Call*, *Disconnect Terminating Call*, *Distinguish Call Scenarios*, and *Handle Traffic* Use Cases. Also, The Use Case diagram illustrates the RANAP Controller functionalities which are interpreted into the *Attach SCCP*, *Detach SCCP*, *Check SCCP Service*, *Forward Call Scenario Messages*, and *Forward CPP Response Messages* Use Cases. Furthermore, the Use Case diagram explains all Use Cases that provide the SCCP Interface Controller functionalities, these Use Cases are: the *Control CPP Interfacing*, *Disconnect SCCP Connection*, *Setup SCCPApfi Service*, and *Release SCCPApfi Service* Use Cases. On the other hand, the CPP and the RanapSim Manager component are represented as UML Actors in the Use Case diagram. More details about the RanapSim Use Case Model can be found in [23].

B. UML Analysis Model

The Analysis Model describes the logical implementation of the functional requirements that we identified in the Use Case Model. The main purposes of the Analysis Model are to: (1) identify the classes which perform a Use Case's flow of events, (2) distribute the Use Case behavior to those classes through Use Case realizations, (3) identify the responsibilities, attributes and associations of the classes, and (4) build the Class Diagrams of the system [2].

1) Analysis Classes: They represent an early conceptual model of the system, the class is a description of a set of objects that share the same attributes, operations, relationships, and semantics. For the *RanapSim* analysis model, we have proposed several analysis classes to represent data, control, and interfacing functionalities that can be summarized as follows:

- RANAP/CC/MM Message Proxies (32 control and entity classes).
- Traffic Handler/RANAP Procedures (11 control classes).
- SCCP Interface Controller (2 boundary and control classes).
- RANAP Controller (2 boundary and control classes).

More details about the RanapSim analysis classes can be found in [23].

2) Use Case Realization: Here, we have all possible flows or scenarios modeled into sequence diagrams. Sequence diagrams describe the logical implementation of the functional specifications that we identified in the Use Case model. Each Use Case in the RanapSim Use Case Model has a scenario realized through a sequence diagram; all of these diagrams are realized and explained in [23]. For instance, the *Update Location* Use Case represents the location update call scenario through the sequence diagram; Figure 6 shows the main flow of the location update sequence diagram.

The location update procedure is used to update the registration of the actual location area of a mobile station in the network [4], the *Update Location* Use Case handles the traffic generation of the location update call scenarios for any request received from the Manager. This entity should
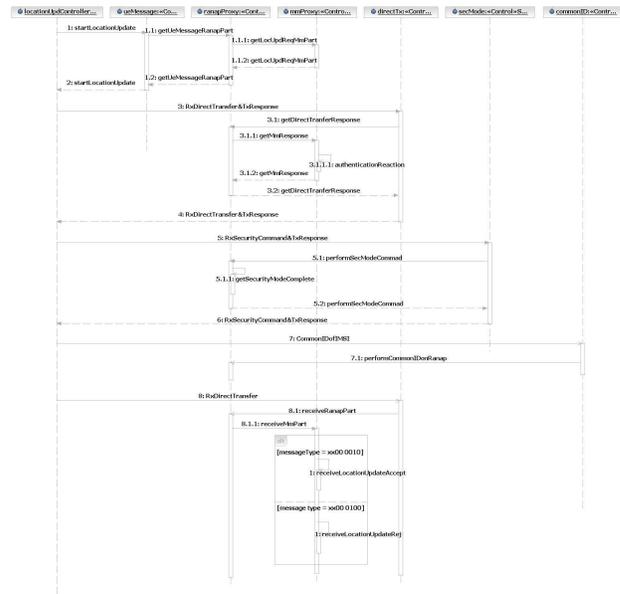

Fig. 6. Location Update Sequence Diagram

model the RANAP and the MM protocols' messages; which is normally carried by the IuCS interface between the RNC and the MSC.

The realization of this Use Case shows the initiation of the *Location Update* call scenario by sending the request message to the MSC; which will emulate the mobile request for location update. In addition, the realization of this Use Case authenticates the connection with the MSC and performs other security procedures to secure and cipher the connection with the MSC, all these procedures are usually done by the mobile station which updates its location. After that, the MSC sends a message that contains the International Mobile Subscriber Identity (IMSI), this message will be received by the sequence diagram to attach it to the emulated mobile station for identification purposes.

3) Class Diagram: This phase of the Analysis Model shows a collection of declarative model elements, such as classes, interfaces, and relationships. It is possible to use Class Diagrams to model the objects that make up the system, to display the relationships between the objects, and to describe what services are provided by those objects [2].

The RanapSim Analysis Model has 9 class diagrams: Location Update, Mobile Originating Call, Mobile Terminating Call, Traffic Handling, RANAP Message, Mobility Management Message, Call Control Message, Control RANAP, and Interfacing for SCCP. Figure 7 shows the UML representation for the Location Update class diagram.

In this class diagram, the *Location Update Controller* class uses the *IRanapProcedures_LocUpd* Interface (Figure 7) through a *dependency/use* relationship [2]. This interface is implemented through an *interface realization* relationship by the Direct Transfer Controller, Security Mode Controller, Common ID Controller, and Initial UE Message Controller classes. Those classes use the *IRanapMessage_LocUpd* Interface (Figure 7) which is realized or implemented by the RANAP message proxy class, through this interface, the
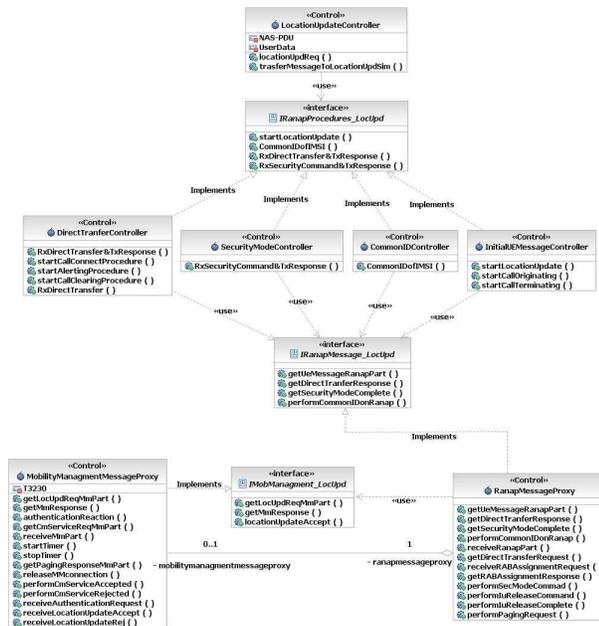
Fig. 7. Location Update Class Diagram

RANAP message proxy class is able to specify the required operations to provide them to other classes.

The RANAP message proxy class uses some operations from the Mobility Management message proxy class which provides these operations through the *IMobManagment_LocUpd* Interface. The RANAP message proxy class has an *association/aggregation* relationship [2] with the MM message proxy class with (1 - 0..1) multiplicity. In this context, this notation means that one object of the RANAP message proxy may have zero or one object of the MM message proxy. Based on the used RANAP procedure, some of the RANAP messages may have a MM message inside. More information about other class diagrams can be found in [23].

VI. CONCLUSIONS

In this paper, we have provided an efficient technique to generate signaling messages to load the MSC server in the UMTS networks in order to exercise load testing. We have modeled the Radio Access Network Application Part protocol's procedures in order to generate the radio access network messages in the IuCS Interface. Also, we have modeled the mobile radio interface protocols in the UMTS network, including the Mobility Management and the Call Control protocols.

The traffic generator, called *RanapSim*, has been modeled using the UML Use Case Model and the Analysis Model, through which we have modeled the connectivity functionality for the traffic generator design to communicate with Ericsson's Connectivity Packet Platform (CPP) through the SCCP protocol. As a future work, the traffic distribution in a multi-processor application can be modeled to generate traffic similar to multiple RNCs traffic to have extremely high traffic load generation.

REFERENCES

[1] IBM Corporation. Rational Systems Developer Tool. http://www.ibm.com, 2007.
[2] J. Rumbaugh, I. Jacobson, and G. Booch. *The Unified Modeling Language Reference Manual*. Addison-Wesley, 2005.
[3] 3GPP TS 25.413: Universal Mobile Telecommunications System (UMTS); UTRAN Iu Interface Radio Access Network Application Part (RANAP) Signaling, version 5.12.0 Release 5, 2005.
[4] 3GPP TS 24.008: Digital Cellular Telecommunications System; Universal Mobile Telecommunications System (UMTS); Mobile radio interface Layer 3 Specification; Core Network Protocols; Stage 3, version 5.16.0 Release 5, 2006.
[5] 3GPP TS 25.412: Universal Mobile Telecommunications System (UMTS); UTRAN Iu Interface Signaling Transport; version 5.2.0 Release 5, 2004.
[6] ITU-T Recommendation Q.711: Language and General Doftware Aspects for Telecommunication Systems, Specification and Description Language (SDL), 1992.
[7] ITU-T Recommendation Z.120: Language and General Software Aspects for Telecommunication Systems, Message Sequence Charts (MSC), 1993.
[8] Ericsson Inc. CPP Platform Specification, SCCP Signaling Service Using UML, Ericsson confidentainal information, 2007.
[9] Ericsson Inc. MGwSim Simulator Architecture, Ericsson confidentainal information, 2005.
[10] Ericsson Inc. 3Gsim Solution, Ericsson confidential information, 2008.
[11] Ericsson Inc. UTMS Solution, Ericsson confidential information, 2006.
[12] B. Reed. Global Telecom Revenue to Hit $2 Trillion in '08, Network World, http://www.networkworld.com/news/2008/091708-global-telecom.html.
[13] J. Lee and P. Hsu. Design and Implementation of the SNMP Agents for Remote Monitoring and Control via UML and Petri Nets. *IEEE Transactions on Control Systems Technology*. 12 (2): 293-302, 2004.
[14] Catapult Communications. MGTS System Reference Manual, 2007.
[15] Catapult Communications. DCT2000 System Reference Manual, 2007.
[16] Polystar Inc. SOLVER System Information, 2007.
[17] D. Loukatos, L. Sarakis,K. Kontovasilis, C. Skianis, and G. Kormentzas. Tools and Practices for Measurement-based Network Performance Evaluation. In Proceedings *Personal, Indoor and Mobile Radio Communications*, pp. 1-5, Athens, Greece, 2007.
[18] D. Loukatos, L. Sarakis, K. Kontovasilis, and N. Mitrou. An Efficient ATM Traffic Generator for the Real-Time Production of a Large Class of Complex Traffic Profiles. *Journal of Communication and Networks*. 7 (1): 54-64, 2005.
[19] F. Sandu, S. Cserey, I. Szekely, D. Robu, and T. Balan. Simulation of an Advanced Mobile Communication Network. In Proceedings *Optimization of Electrical and Electronic Equipment*, pp. 223-230, Brasov, Romania, 2008.
[20] A. Varga. OMNeT++ Discrete Event Simulation System Version 3.2 User Manual, 2005.
[21] M. Mackaya, O. Kone, and R. Castanet, Modeling Location Operations in UMTS Networks. In Proceedings *Modeling Analysis and Simulation of Wireless and Mobile Systems*, pp. 69-73, Atlanta, Georgia, USA, 2002.
[22] S. Rios. Location Based Services: Interfacing to a Mobile Positioning Center. http://www.wirelessdevnet.com, 2009.
[23] S. Albasheir: *Modeling of Radio Access Application Protocols for Mobile Network Traffic Generation*. M.A.Sc Thesis, Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada, 2008.