Orientation towards a Service Oriented Architecture for designing and building a Comprehensive Emergency System

Hamdan Al-Sabri, Samir El-Masri

**Department of Information Systems, College of Computer and Information Sciences, King Saud University
Riyadh, Saudi Arabia**

Abstract

As a result of the Revolution in the information technology, software and telecommunication and the increase of the volume of data, process of storing and retrieving data in emergency systems; it becomes necessary to work on the integration between all applications involved in emergency to produce a Comprehensive emergency system. Service Oriented Architecture (SOA) is an ideal solution to resolve integration and interoperability issues between independent and heterogeneous systems by relying on the concept of Web Services/Mobile Web Services. Most Emergency systems are not comprehensive and not built using modern technologies. In this paper, we deal with this challenge by using Mobile web service and Service oriented architecture to modeling and designing a service oriented framework for Comprehensive Emergency System (CES). The proposed CES is composed of the following distributed subsystems: emergency application for mobile devices; Global Positioning System (GPS); Ambulance system, Main central system, Health record and Hospital System. The various components of the CES work together to handle all operations from emergency request until the patient's treatment at an emergency department. The overall behaviors of the system are explained by extracting all services that provide or consume by components and cooperation among them.

Keywords: *Service Oriented Architecture (SOA), Service oriented architecture Modeling Language (SoaML), Comprehensive Emergency System (CES), Mobile Web Services.*

1. Introduction

Service Oriented Architecture (SOA) is a pattern or approach that provides a guidance to extract functions of large systems and transform the system functionalities into services through all stages of the life cycle of the system. SOA is also a form of distributed systems that supports a cloud computing, and an integration between applications and exchange data via services in an easy and flexible manner regardless of the platforms, and programming languages used to build the applications [1], [29]. Web Services technology is one of the methods used to apply SOA approach through integration of web-based applications. Web Services are used by many applications regardless of software environment and how to communicate with each other and the type of the host of applications [2], [3]. Web services technology is a standardized way of integrating web-based applications. Web services communicate using open protocols, and based on new standards eXtensible Markup Language (XML), Simple Object Access Protocol (SOAP), Web Service Description Language (WSDL) and Universal Description Discovery and Integration (UDDI).

A service is a location transparent where consumers do not need to be aware of the physical location of a hosting server. It is also a protocol independent, where messages are sent in a platform-neutral and standardized format delivered through the service specification. Services tend to be oriented toward use over a network, though this is not an absolute requirement [13] .The growing need to process of access to applications by mobile device leads us to a new technology called Mobile Web Services Which has been initiated by El-Masri (2005) [4], El-Masri and Solumein (2005) [5] and El-Sappagh et al [31].

The main advantage of Mobile Web Services is the communication between mobile subsystems [28], [30]. Wireless Application Protocol (WAP) and Wireless Markup Language (WML) standards are necessary to build Mobile Web Services, where WAP is an open standard to enable mobile device user to easily access information and services through wireless connection and vice versa WML has also the ability to provide users with visual interfaces [6], [7], [8].

In this paper, we present the use of SOA, SoaML principles to identify and extract services for CES). Services and the framework for communication between CES components will be also designed, presented and discussed. CES has sufficient flexibility where all system components can communicate among them using Mobile Web Services technology. The system is more comprehensive than other existing emergency systems, due to a range of benefits: CES uses modern techniques in the process of integration between system components; Selects the appropriate ambulance, doctor and appropriate treatment according to patient diagnosis. More details and more features will be shown in next sections.

This paper is organized as follows; Section 2 will present background and related work and overview SOA and model techniques. In section 3, the proposed architecture for CES and their components will be explained. In section 4, the distribution for communication between CES subsystems will be shown. Services modeling will presented in section 5 and XML contract between CES components is discussed in section 6. Conclusions and future research work will conclude the paper in section 7.

2. Background and Related Work

The SoaML Metamodel extends the Unified Modeling Language 2.0 (UML) in order to support SOA. This section only presents the concepts of SoaML which are based on the revised UML Profile and Metamodel for Services (UPMS) submission presented in the literature [11], [12], [16]. SoaML allows modeling the system as services architecture, represented by UML collaboration diagrams. This services architecture provides a high-level and contextual view of the roles of the participants and services [9], [10].

Services, service orientation, and composition are the important concepts in SOA. To build services in SOA approach, it is necessary to identify candidate services, specify exposed services, and make realization decisions to design the SOA solution [14], [15]. Business Process models the flow of tasks, the business items that are passed between tasks, and the roles that perform these tasks; it also describes where automation is required. Business Goals requirements must be met by the business and Information Model information (data) that might be exposed through services [16], [17].

Candidate services are a set of functions that a service provider by one or more participants might offer. A participant plays the role of service provider when offers services or service consumer uses services. [18] Presents an approach called Ambient SoaML, which introduces ambient in service oriented architecture modeling language (SoaML). It explains the simple mobility and demonstrates the use of SoaML for modeling SOA of a mobile application.

A new model-driven approach is introduced for the generic integration of service-oriented architectures (SOA) and multi-agent systems (MAS)[19]. In another paper, MINERVA was proposed for automating transformations from BPMN to SoaML models in order to automatically generate services from business processes [20]. In [21], the authors analyzed common and widespread service characteristics, derive evaluable design attributes that refer to elements of service designs based on SoaML, and demonstrate the formalization of an exemplarily design attribute using OCL.

Another approach was presented with an example of service identification from the Norwegian national Health ICT architecture by using SoaML [22]. [23] Shows how SOA modeling and design based on the concept of service component and standard UML modeling constructs and defines service components of different types, scope and granularity. It puts them in the context of a model-driven design approach to provide bidirectional traceability between business requirements and software artifacts.

Paper [24] showed the use of shared data models of emergency incidents to support the exchange of data between heterogeneous systems. Summarizes found in [25] on how to use service oriented architecture to lightweight mobile devices. In [26] and [27], the authors focused on investigating the importance of data exchange and message passing on SOA from the security and privacy point of view.

Thereafter, they designed a gateway for passing messages in the SOA healthcare platform. Subsequently, they pointed out the interface utilities on the SOA healthcare platform. Healthcare information integration and shared platform based on Service-Oriented Architecture (SOA) was proposed. The platform supports the integration, development, and operation of a full spectrum of healthcare applications.

3. The proposed architecture for a CES

A Comprehensive Emergency System (CES) is a comprehensive environment working to link several components to perform an integrated task. The components are mobile device application, main central system, ambulance system, health record system and hospital system. All these components work to provide high quality service and facilitate the transfer of data from the beginning of the reporting of accident then request ambulance, patient historical data and provide treatment for the patient. The system facilitates communication between those components that are linked through the use of the Service Oriented Architecture/Mobile Web Services (SOA/MWS). The accident reporter (Emergency requester) device and ambulance systems play the role of mobile web service providers. The abstract view of SOA/MWS for CES illustrates the CES as a black box with inputs (accident information, available ambulance coordinate, patient’s ID, current condition of the patient) and outputs (report for emergency requester, choose the right ambulance, confirm take/reject case, patient’s health record, choose the appropriate treatment, choose the appropriate hospital and contact medical consultant) as shown in Figure 1.

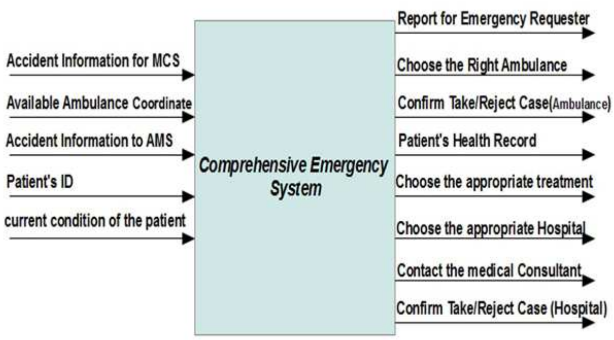

Fig. 1 The abstract view of the CES architecture.

The comprehensive Emergency System has the following subsystems that they interact together: Mobile Device Application, Main Central System, Ambulance Systems, Electronic Health Record System and Hospital Emergency Systems, as shown in Figure 2. We will summarize the roles of the subsystems as follows:

1) The *Mobile Device Application (MDA)* System:
   - sends emergency message to the MCS,
   - receives the report from MCS.

2) The Main Central System (MCS):

- receives emergency message
- requests the location of available calculates the distances between available ambulances and the accident place, and choose the closest
- sends request to the right chosen ambulance to take the job sends report to the initial caller in a few seconds reporting the distance and time to reach the accident
- is able to receive normal emergency call on 997.

3) The Ambulance System (AMS):

- has capability of reading from GPS location.
- sends ambulance location to the MCS and informs if vehicle is available, non-available or in mission.
- sends the confirmation take/reject job to MCS.
- access the health record of the patient
- can propose the appropriate treatment of the patient through a decision support system.
- can determine the suitable hospital and send patient information to the hospital system
- is able to contact the medical consultant.

4) The Hospital System (HS):

- receives the patient's information from AMS.
- can request the patient's health record.
- checks the availability of beds and doctors in hospital.
- sends the confirm take/reject case.
- can call for consultants.
- can call for consultants.

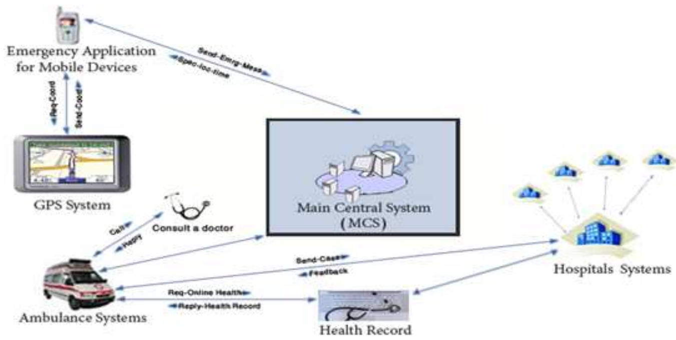

Fig. 2 The detailed view of the architecture for CES

4. The distribution communications between CES subsystems

SOA is a form of distributed systems architecture based on services where a consumer does not need to know the internal structure of a provided service, including features such as its implementation language, process structure, and even database structure, for that, we suggested the three servers for communications between CES subsystems and other servers for replication as shown in Figure 3. We suggest the server for Emergency Mobile Device Application, Main Central System, and server for Ambulance System and server for Hospital System with health records and use the distributions models.

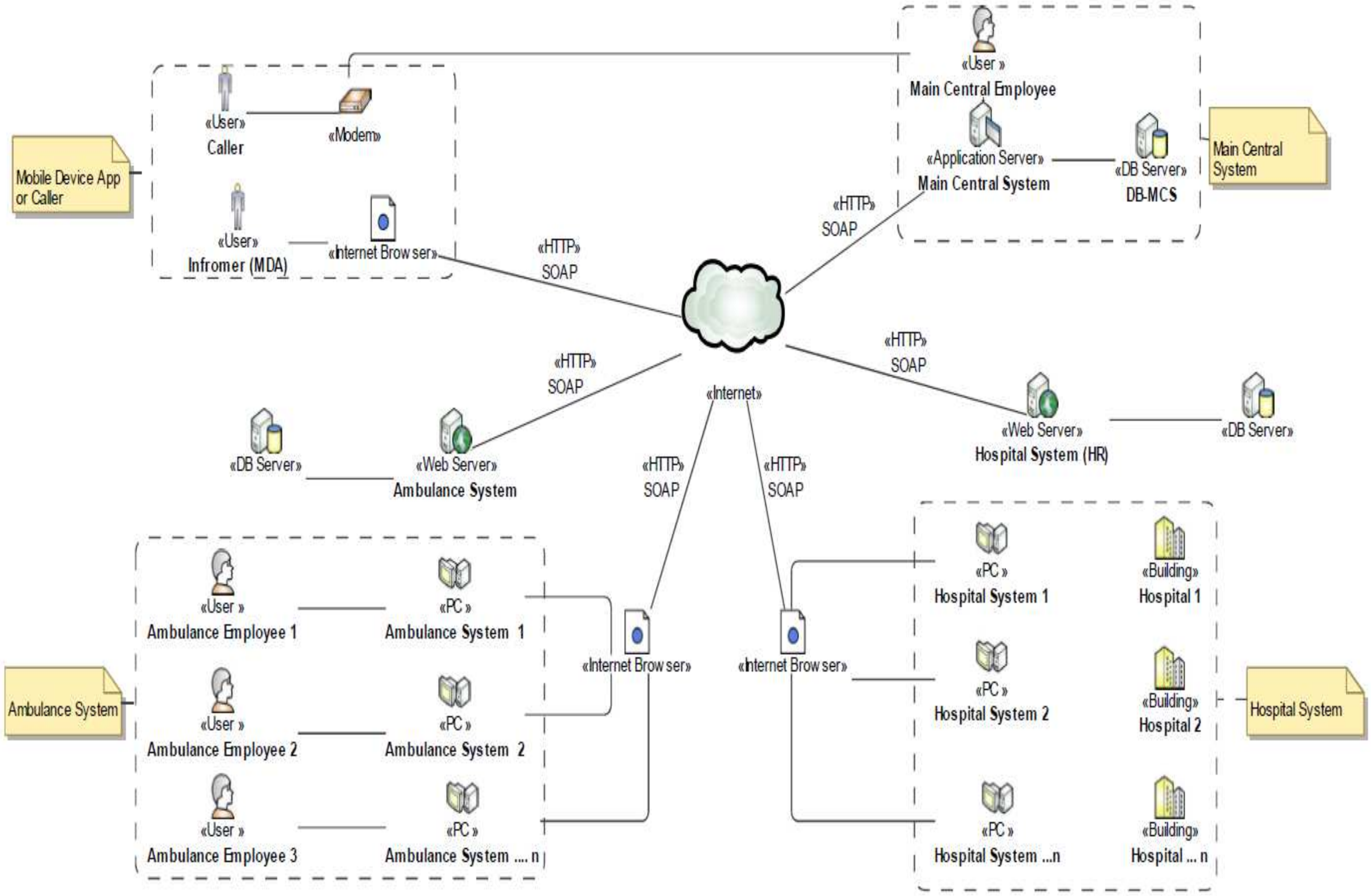

Fig. 3 CES subsystems connectivity simulation

5. Services Modeling for CES

In this section, we use the SoaML that allows modeling this community as services architecture, represented by UML collaboration. This services architecture provides a high-level and contextual view of the roles of the participants and services. The Comprehensive Emergency System network involves four primary roles for participants (Emergency Mobil Device Application, Main Central System, Ambulance System, and Hospital System). In this community, the participants participate in nine business-to-business (B2B) services (*Emergency Event, Handling Emergency Event, Ambulance Role, Ambulance in Action, Best Choose Hospital, Accessing Health Record, Hospital Availability, Hospital in Action, and Choose Consultant*).

5.1 Candidate Services

In the service identification, the Capabilities (Candidate Services) are provided to organize SoaML capabilities and UML elements that represent resources from which capabilities can be identified. Capabilities represent candidate services at this stage. In the Service identification stage, we defined nine candidate services (Emergency Event, Handling Emergency Event, Ambulance Role, Ambulance in Action, Best Choose Hospital, Accessing Health Record, Hospital Availability, Hospital in Action, and Choose Consultant) that interact between CES components, as shown in Figure 5.
Emergency Event Service this service use to interaction between Mobile Device Application, Main Central System and Ambulance System as shown activity diagram in Figure4. It has three operations (send emergency request, receive request with GPS coordinates, and receive request available ambulance); those operations used to send accident information (accident locations, cars numbers, injures numbers, and accident date) and request the available ambulances that are closed to the accident place. Services will be clarified in more details in services collaboration section.

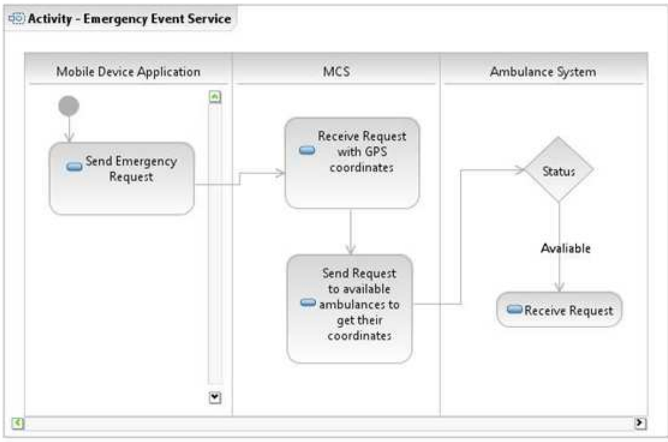

Fig. 4 Activity diagram for Emergency Event Service

5.2 Services Collaboration in CES

Services collaboration Occur between participants and participants represents some, possibly concrete, party or component (people, organizations or systems) that provides services, consumes services, or both. A participant plays the role of service provider if it offers a services or service consumer if it uses services. Services can play the provider role in some services and consumer in others. In the Comprehensive Emergency System there are nine services interactions between CES components (Mobile Device Application, Main Central System, Ambulance System, and Hospital System). We explain the details of the services on Table 1. And use of services between the components in the form of services collaboration as shown in figure 5.

Table 1: Details of Comprehensive Emergency System Services

Service Name	***Service Description***
Emergency Event Service	This service used to interact between Mobile Device Application, Main Central System and Ambulance System. It has three operations to send accident information and request the available ambulances that close from accident place.
Handling Emergency Event Service	This service use to interaction between Main Central System and Ambulance System. It has four operations use to calculate distance between ambulances locations and accident location and compare between retrieve coordinates and determine the best ambulance that are closed to the accident place then it sends job request to ambulance.
Ambulance	This service interacts between Main Central System, Ambulance System, and Mobile

Role Service	Device Application. It has four operations that send accident information by Main Central System for Ambulance System.
Ambulance in Action Service	This service just used by the Ambulance System to modify ambulance status from available to on-mission. It has three operations used by ambulance.
Best Choose Hospital Service	This service used to interact between Ambulance System and Hospital System. It has three operations used after ambulance reached to the accident place, The Ambulance System will request health record for patient by using patient's ID.
Accessing Health Record Service	This service used by Ambulance System and Hospital System to request Electronic Health Record for patient.
Hospital Availability Service	This service used to interact between Ambulance System and Hospital System. It has four operations to get patient historical data, get current attribute patient from ambulance, check available beds, and check available staff/doctors.
Hospital in Action Service	This service used to interact between Ambulance System and Hospital System. It has three operations to receive patient from ambulance, alter ambulance status to available, and process diagnosis.
Choose Consultant Service	This service used by Ambulance System and Hospital System just to request consultant information.

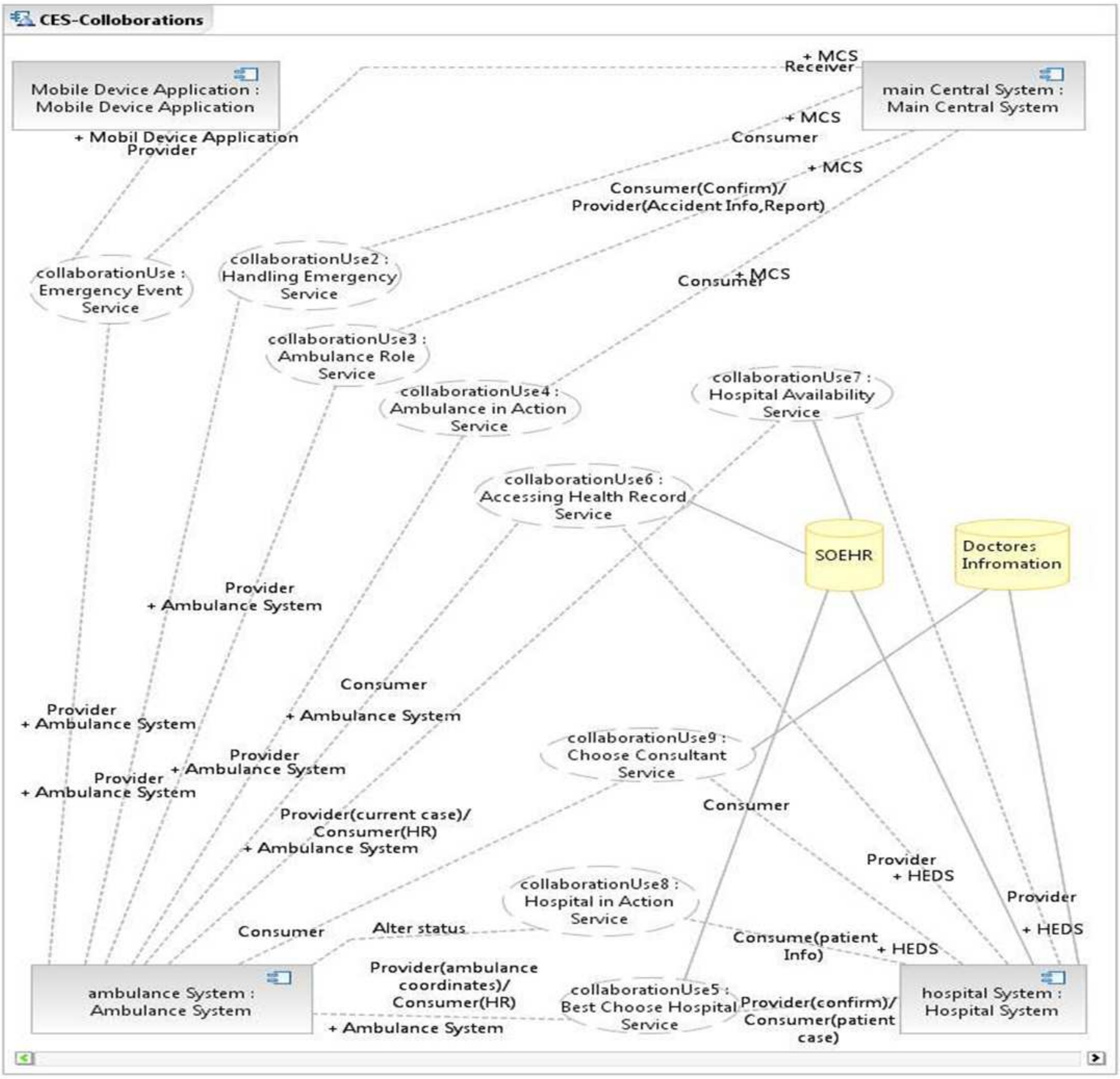

Fig. 5 Service Collaboration diagram for Comprehensive Emergency System

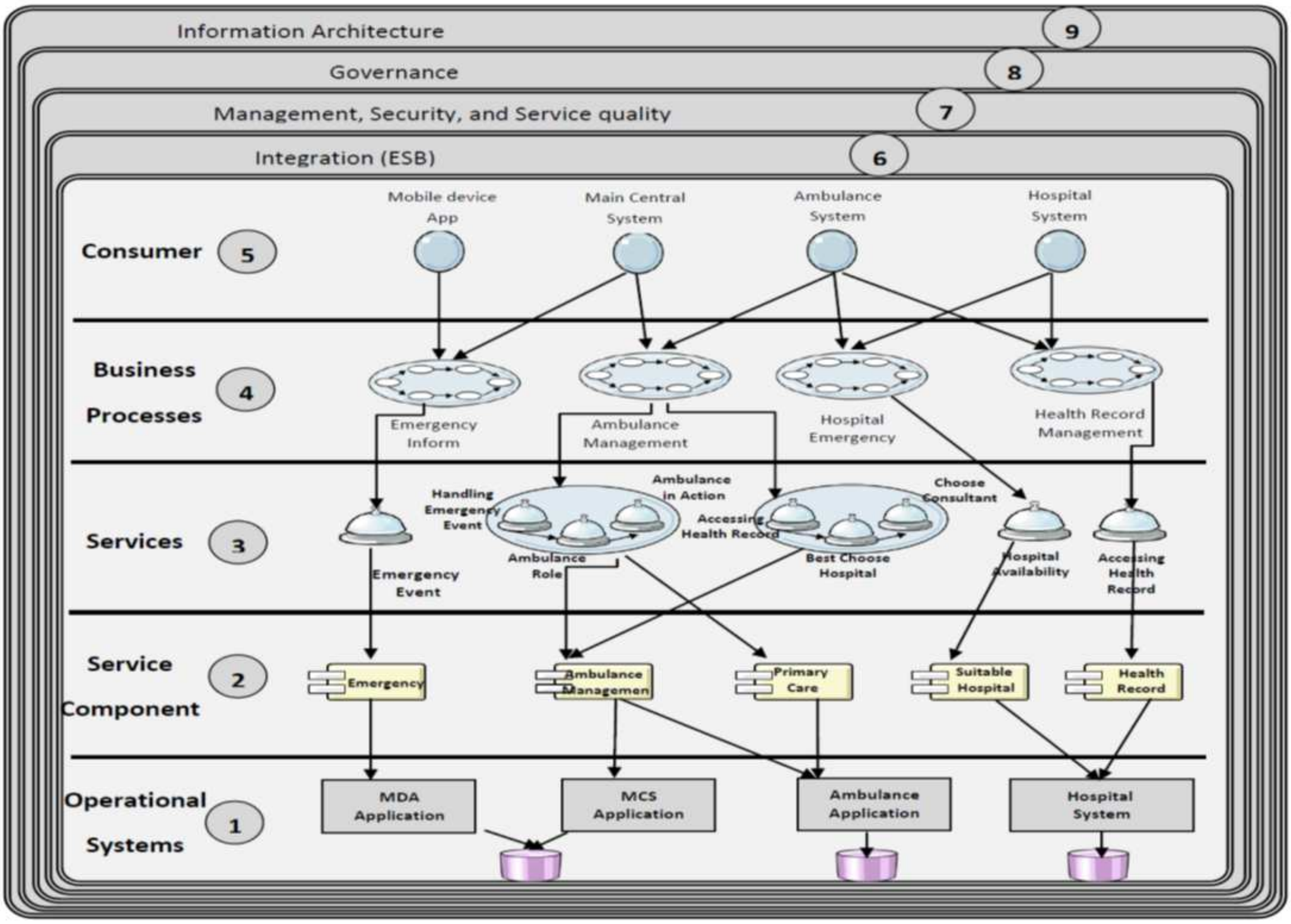

Fig. 6 SOA reference Architecture for a CES

5.3 SOA architecture for CES

An abstract view of SOA layers showing the process of communication between all layers in order to complete the integrate work for a Comprehensive Emergency System will be presented. This layered architectural approach guides the adoption of SOA. The relationship between services and components is those components (Mobile device application, Main Central System, Ambulance System, Hospitals System with Health Records) that implement the services, and responsible for providing their functionalities, and maintaining their quality of service. Business process flows can be supported by choreography of these exposed services. Integration architecture supports the routing, mediation, translation of these services, components, and flows using an Enterprise Service Bus (ESB). In this section we will focus on just five layers linking the components and services to achieve desired business goals as following:

Operational systems represent existing IT assets, and shows which IT investments are valuable and can be leveraged in an SOA. In Comprehensive Emergency System there are four IT assets (Mobile Device, Main Central System, Ambulance System, and Hospitals System).

Service components realize services, possibly by using one or more applications in the operational systems layer. In the Comprehensive Emergency System there are five service components (The possibility of request emergency, ambulance management, provide primary care, choose suitable hospital and request health record for patient) used the services to accomplish the desired tasks from CES.

Services represent the services that have been deployed to the environment as shown previously.

Business Process represents the operational artifacts that implement business processes as choreographies of services. In Comprehensive Emergency System there are four big business processes (Emergency Informer, Ambulance Management, Hospital Emergency, and Health Record Management).

Consumers represent the channels that are used to access business processes, services, and applications. In Comprehensive Emergency System there are four consumers requesting the services by (Mobile Device Application, Main Central System, Ambulance System, and Hospital System). All layers mentioned above that show SOA architecture for a Comprehensive Emergency System shown in Figure 6.

6. XML contract between CES components

The CES components are classified into consumer, provider or both roles as shown in Figure 5. All these components cooperate and share data using the standard WSDL that is based on XML. SOAP is an XML-based protocol from the W3C for exchanging data over HTTP. Web services use SOAP to send messages between a service and its client(s). SOAP messages are XML documents that contain: Envelope that encloses the message itself; Header (optional); Body that includes the message payload and Fault (optional). The contract between Mobile Device Application and Main Central System that specifies accident information is shown in table 2. The report for informer that is sent from Main Central System to Mobile Device Application and specifies some information about ambulance that reaches accident place is shown in table 3.

Table2: Contract structure for Accident Information

```
<? xml version="1.0" encoding="utf-8"?>
  <Accident-Information>
      <Accident-location>
            <Street-no> 12 </Street-no>
             <Area-no> 1 </Area-no>
             <Coordinates>335.221, 3225.200 </Coordinates>
     </ Accident-location>
      <Cars-no>   2 </Cars-no>
      <injured-no> 3 </injured-no>
  </Accident-Information>
  <Report-to-emergency-requester>
     <Time-to-reach> 10 minutes </Time-to-reach>
    <Distance> 3.5 KM </Distance>
 </Report-to-emergency-requester >
</xml>
```

Table3: Contract structure for Emergence Request

```
<? xml version="1.0" encoding="utf-8"?>
  <Report to Informer>
      <Ambulance-location>     (2335.2,899,558)
      </ Ambulance-location>
      <Time to reach> 12m:20s </Time to reach>
      <Distance> 6 KM </Distance>
      <Date> 12/5/2011 </Date>
  </Report for Informer >
</xml>
```

7. Conclusions and further work

This paper presented the architecture for a Comprehensive Emergency System (CES) by using Service Oriented Architecture/ Service Oriented Architecture Modeling Language (SOA/SoaML). In this comprehensive system we used a suitable technology (Mobile Web Services) to facilitate the exchange data between CES components and give the system sufficient flexibility to communicate from anywhere. The CES subsystems and their roles have been identified and simulated. The integrated work flow between the CES subsystems and the way to share data has been presented.
As for a future work, we plan to use the quality of services (QoS) to test the performance of all subsystems with the CES framework. Also, we plan to apply how to separately control the unexpected events and the subsystems rules from the separated engines. Moreover, we plan to use scripting to model the rules within these engines.

Acknowledgements

This work is part of a two year research project which has been fully funded by a grant through King Abdul-Aziz City for Science and Technology (KACST) / National Plan for Science and Technology (NPST) in the Kingdom of Saudi Arabia. Grant number: 09-INF880-02.

References

[1] Gurpreet Singh Modi,"Seminar Report on Service Oriented Architecture & Web 2.0,"2004.
[2] The World Wide Web consortium W3C, available at: http://www.w3.org/. [Online]. Access date: March 30, 2005.
[3] Web Services Interoperability, available at: http://www.ws-i.org/. [Online]. Access date: March 30, 2005.
[4] Samir El-Masri,"A Framework For Providing Mobile Web Services," School of Computing and IT, University of Western Sydney, Australia.
[5] Samir El-Masri and Basem Suleiman,"A Framework for Providing Mobile Web Services", The second International Conference on Innovations in Information Technology (IIT'05) Dubai, UAE – September 26 – 28, 2005.
[6] Patil, B., Saifullah, Y., Faccin, S., Sreemanthula, S., Aravamudhan, L., Sharma, S. and R. Mononen, IP in Wireless Networks, Prentice Hall PTRP, January 31, 2003.
[7] Wireless Application Protocol White Paper, available at: http://www.wapforum.org/what/WAP_white_pages.pdf. [Online]. Access date: April 5, 2005.
[8] Wireless Markup Language Specification Version 1.1, available at: http://www.wapforum.org/what/technical.htm. [Online]. Access date: April 5, 2005.
[9] Brown, Alan et al. "SOA Development Using the IBM Rational Software Development Platform: A Practical Guide." IBM.com ftp://ftp.software.ibm.com/software/rational/web/whitepapers/G507-0956-00.pdf
[10] Wahli, Ueli et al. Building SOA Solutions Using the Rational SDP. IBM Redbooks, sg247356, 2007.
[11] Service oriented architecture Modeling Language (SOAML) MDA OMG - Object Modeling Group, 2008.
[12] A Berre, "Service oriented architecture modeling language (SoaML)-specification for the UML profile and metamodel for services (UPMS)" Object Management Group (OMG), 2008.
[13] World Wide Web Consortium, http://www.w3.org/.
[14] Arsanjani, Ali et al. "Design an SOA Solution Using a Reference Architecture." IBM developerWorks, 2007.http://www.ibm.com/developerworks/library/ar-archtemp/index.html.
[15] Booch, Grady. "Snake Oil-oriented Architecture."IBM developerWorks, 2006.http://www-03.ibm.com/developerworks/blogs/page/gradybooch?entry=snake_oil_oriented_architecture.
[16] OMG. Documents Associated with Service oriented architecture Modeling Language (SoaML). http://www.omg.org/spec/SoaML/Current
[17] Lawrence, Calvin. "Adapting Legacy Systems for SOA."IBM developerWorks, 2007. http://www-128.ibm.com/developerworks/webservices/library/ws-soa-adaptleg/
[18] Ali, N.; Babar, M.A., "Modeling Service Oriented Architectures of Mobile Applications by Extending SoaML with Ambients," 35th Euromicro Conference on Software Engineering and Advanced Applications, Petras, Greece August 27-29, 2009.
[19] Hahn, C.; Jacobi, S.; Raber, D., "Enhancing the Interoperability between Multiagent Systems and Service-Oriented Architectures through a Model-Driven Approach", IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010.
[20] Delgado, A.; García-Rodríguez de Guzman, I.; Ruiz, F.; Piattini, M., "From BPMN business process models to SoaML service models: A transformation-driven approach,", 2nd International Conference on Software Technology and Engineering (ICSTE), 2010.
[21] Gebhart, M.; Baumgartner, M.; Oehlert, S.; Blersch, M.; Abeck, S.; , "Evaluation of Service Designs Based on SoaML," Fifth International Conference on Software Engineering Advances, Nice, France, August 22-27, 2010
[22] Fenglin Han; Moller, E.; Berre, A.J, "Organizational Interoperability Supported through Goal Alignment with BMM and Service Collaboration with SoaML", International Conference on Interoperability for Enterprise Software and Applications, Beijing China, April 21-21, 2009.
[23] Stojanovic, Z.; Dahanayake, A.; Sol, H;, "Modeling and design of service-oriented architecture,", IEEE International Conference on Systems, Man and Cybernetics, 2004.
[24] Hauenstein, L.; Tia Gao; TszWoSze; Crawford, D.; Alm, A.; White, D.; , "A Cross-Functional Service-Oriented

Architecture to Support Real-Time Information Exchange in Emergency Medical Response," Engineering in Medicine and Biology Society, 28th Annual International Conference of the IEEE, vol.Supplement, no., pp.6478-6481, Aug. 30 – Sept 3 2006.

[25] Tergujeff, R.; Haajanen, J.; Leppanen, J.; Toivonen, S.; , "Mobile SOA: Service Orientation on Lightweight Mobile Devices", IEEE International Conference onWeb Services, vol., no., pp.1224-1225, 9-13 July 2007

[26] Chi-Lu Yang; Yeim-Kuan Chang; Chih-Ping Chu; , "A Gateway Design for Message Passing on the SOA Healthcare Platform", IEEE International Symposium onService-Oriented System Engineering, vol., no., pp.178-183, 18-19 Dec. 2008.

[27] Yong-Gang Gong; Xin Chen;, "Healthcare information integration and shared platform based on service-oriented architectures", 2nd International Conference onSignal Processing Systems (ICSPS), vol.2, no., pp.V2-523-V2-527, 5-7 July 2010.

[28] Samir El-Masri, "mobile comprehensive emergency system using mobile web services". The second International Conference on Innovations in Information Technology (IIT'05) Dubai, UAE,September 26 – 28, 2005.

[29] Krafzig, D. Banke, K. Slama, D., "Enterprise SOA, Service Oriented Architecture: Best Practices", Prentice Hall, 2005.

[30] Yuri, Huaigu, Yi Zheng.; "Service-Oriented Mobile Applications for Ad-Hoc networks", 2008 IEEE International Conference on Services Computing.

[31] Shaker H El-Sappagh, Samir El-Masri, Alaa M. Riad and Mohammed Elmogy, "Electronic Health Record Data Model Optimized for Knowledge Discovery", International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012.

Hamdan Al-Sabri received his B.Sc. degree in computer science in 2009 and his Master degree from King Saud University in 2011, and he is currently doing his PhD at the department of information Systems, college of computer and information sciences, King Saud University, Saudi Arabia. Al-Sabri has published a few papers in SOA.

Dr Samir El-Masri completed his Electrical and Electronic Engineering degree at the College of Engineering, Lebanese University in 1993. Dr El-Masri has a Master's degree and PhD from the "Institut National Polytechnique de Grenoble" France in numerical simulations and software engineering in 1997. He worked for 3 years ended in 2001 at Hokkaido University, Japan as postdoctoral fellow and assistant professor. Dr El-Masri was a senior Lecturer/Associate Professor at the University of Western Sydney, University of Sydney, Central Queensland University and University of Southern Queensland, Australia from 2001 till 2006. He worked for at least 4 years in IT and software development industry at large Australian companies as a senior project manager/Program Manager and senior consultant in Australia. Dr El-Masri is currently an Associate Professor at King Saud University, Riyadh, Saudi Arabia since 2009, and he has published more than 80 research papers in international Journals, books and Conferences. Research Interests are Health Informatics. Dr El-Masri is a member of IEEE and IEEE Computer Society, Lebanese Order of Engineers, Saudi Association for Health Informatics and Australian Computer Society. He is an editorial member of the journal of engineering and technology research and the journal of health informatics in developing countries.