

System Interoperability Study for Healthcare Information System with Web Services

J.K. Zhang, W. Xu, D. Ewins

Centre for Biomedical Engineering, School of Engineering, Surrey University, UK

Abstract: This paper describes the use of a new distributed middleware technology ‘Web Services’ in the proposed Healthcare Information System (HIS) to address the issue of system interoperability raised from existing Healthcare Information systems. With the development of HISs, hospitals and healthcare institutes have been building their own HISs for processing massive healthcare data, such as, systems built up for hospitals under the NHS (National Health Service) to manage patients’ records. Nowadays many healthcare providers are willing to integrate their systems’ functions and data for information sharing. This has raised concerns in data transmission, data security and network limitation. Among these issues, system and language interoperability are one of most obvious issues since data and application integration is not an easy task due to differences in programming languages, system platforms, Database Management Systems (DBMS) used within different systems. As a new distributed middleware technology, Web service brings an ideal solution to the issue of system and language interoperability. Web service has been approved to be very successful in many commercial applications (e.g. Amazon.com, Dell computer, etc.), however it is different to healthcare information system. As the result, Web Service-based Integrated Healthcare Information System (WSIHIS) is proposed to address the interoperability issue of existing HISs but also to introduce this new technology into the healthcare environment.

Key words: Web service, XML, SOAP, WSDL, UDDI, healthcare information system

INTRODUCTION

With the evolution of IT technologies and increasing demands from people for computerising medical-related information, healthcare institutions have developed systems to manage and process the large amount of medical information. In order to achieve information exchanging and medical knowledge sharing, hospitals and healthcare providers have intended to integrate their systems’ functions and data. This raised some concerns, such as, data security, data transmission, network limitation and so on^[1]. Among these issues, the issue of system and language interoperability are most obvious barrier for the integration of functions and data of different systems. In practice, systems were developed using different languages (e.g. Java, Visual Basic, C++, etc.), different system platforms (e.g. Microsoft Windows operating systems, Linux operating system, etc.) and DBMSs (e.g. Microsoft SQL server, Oracle, Microsoft Access, etc.)^[2-5]. These differences between systems are major factors leading to the issue of system and language interoperability.

Traditionally, developers use distributed middleware technologies, like CORBA (Common

Object Request Broker Architecture)/DCOM (Distributed Component Object Model) to tackle the issue of system and language interoperability. Regarding to the author’s research, there have been some of HISs, such as I-HER^[3], Hospital Information System^[5], BHS^[4]. But CORBA/DCOM would become unsuitable while they are applied into those systems based on Internet environment. That is because CORBA/DCOM has the shortcoming in the firewall crossing and wireless environment^[6-8]. In addition, CORBA/DCOM is a fairly complex issue, a task requiring special expertise^[9]. Nowadays, Web service as a new distributed middleware technology can not only overcome those shortages from CORBA/DCOM but also to successfully address the issue of system and language interoperability.

Web service has been successfully applied in many commercial applications, such as, Ebay, Amazons, Dell and so on, nevertheless, a few of people and organisations formally deployed this technology into healthcare industries. In this paper, a Healthcare Information System named as ‘Web Service-based Integrated HIS (WSIHIS)’ is proposed and built up with the support of web service and some Microsoft .Net-based technologies to address the issue of system and

language interoperability. The main objective of this paper is to illustrate how Web Service and Microsoft .Net technologies are used to deal with the issue of system and language interoperability. Moreover, system interoperability test was carried to assess whether the web service-based solution is really able to cope with the issue of system and language interoperability.

The issue of interoperability of current HISs: The interoperability can be defined as “the capability with which two or more programs can share and process information irrespective of their implementation language and platform”^[5,10]. Interoperability concerns also rise as one of the most urgent priorities of healthcare information systems^[11,12]. Connecting for health, as a single IT provider for the NHS defined the interoperability issue ‘it is impossible for different healthcare information systems to smoothly, seamlessly, privately communicate information when it is needed to where it is needed’^[1]. Most clinical applications are determined by a huge variety of heterogeneous and independent work places, most of them equipped with specialised clinical hardware^[12]. Usually, different parts of a large-scale telemedicine system might be from various vendors, who use different standards and information formats, such as, National Program for IT (NpfIT)^[13]. These systems are also used by people with different levels of expertise and needs. Once the interoperability problem is solved, the development and maintenance of large telemedicine systems can be streamlined with data reuse, code reuse and application reuse. The following content would explain the interoperability problem from different perspectives^[5,10]:

- * **Database System Interoperability:** Patient records are often located in different database systems^[14]; however, data from different database systems (e.g. Microsoft SQL server, Oracle, Microsoft Access, etc.) cannot be exchanged with each other and used by applications based on different DBMSs. For this issue, some HISs choose to stick their organisations with a single database vendor, however this feature would make those HISs lose the flexibility of data transmission with other HISs.
- * **Language Interoperability:** Usually, different HISs could be developed by different IT providers, such as patient record systems under the NHS system developed by couples of software providers^[15], developers make use of different

programming language to built their HISs (e.g. Java, Visual Basic, C++, C#, etc.). This would cause the reuse and share of applications being very hard because of the incompatibility between different programming languages.

- * **System-platform Interoperability:** System platform interoperability means OS interoperability, but over the last few years the Internet browser has emerged as a platform in itself^[16]. As different HISs could be developed based on different development platforms (e.g. Microsoft Windows Operating systems, Linux Systems, IE web browser, Avant web browser, etc.), this feature would make those HISs only work on some certain system platforms. For example, if a HIS is built up based on windows system platforms that would restrict it to also run on other different platforms (e.g. linux).
- * **Semantic Interoperability:** There are many interoperability problems that are caused by semantic differences. Semantic interoperability assumes that the components of the distributed application will have different meanings of the data. For example, the same term may have different meaning contents in different countries, or different terms could have the same meanings. Such situations are common in medical information systems. Usually human intelligence is required to solve such semantic interoperability issues, however this problem has been beyond the scope of this paper.

The following section would mainly discuss the issue of system and language interoperability based on those systems delivered by the NHS system as a typical example. Within recent years, there is a huge program being delivered by the NHS – National Program For IT (NpfIT). NpfIT was proposed in February 2002, which is aiming to modernise the NHS system with information technologies^[17]. In Oct 2002, this program is formally established, which mainly comprises of various healthcare systems for different purposes. It is no doubtful that the interoperability issue is inevitable for NpfIT because of the following reasons:

- * **Different business processes in different ways**^[18]: There are thousands of hospitals under the NHS, most doing different business processes and having different health systems, which makes NpfIT more complex to interoperate with those various systems.

- * **The requirement for the interoperability between systems delivered by NpFIT:** For example, Electronic Transmission Prescription, as one of systems delivered by NpFIT was originally envisaged as a separate system^[15]. However, this program will also be required to be integrated with another program ‘NHS Care Records Service’^[19].
- * **Dozens of different patient-record systems in use across the NHS:** There are dozens of different patient-record systems in use across the NHS developed by different software providers^[15]. This situation would increase the difficulty for the integration of different records.
- * **Need to interoperate with systems on doctors’ desktops**^[15]: NHS Care Records Service, as one of services of NpFIT will need to interoperate with systems on doctors’ desktops. This would be challenging enough if the systems were built from scratch. NpFIT must rely on many existing hospitals and GP systems being connected to the spine that is a kind of new database system of NHS^[15,20].

- * **Two separate NHS systems between Scotland and England:** Since Scotland and England are using two different NHS systems ^[21,22], both of places are also developing two different Healthcare Information Systems with different IT providers. This situation may lead to the potential system interoperability issue if both systems are expected to be integrated for some certain purposes.

Furthermore, under the NHS system, most existing IT systems in trusts are based on either buildings or departments^[13]. These systems do not usually support the movement of information between buildings and departments. Consequently, within a single organisation, several records are often created for the same patient^[13]. Similarly, in primary care, individual practices have their own IT applications and databases, so patient records are not easily transferred to other practices or care providers^[13].

The solution of interoperability – web service:

Seamless interoperability with applications and data from different systems is a challenging task. Traditionally, solutions of interoperability normally involve developing middleware applications to communicate the non-interoperable applications using the messages, of which the technology is also named as the distributed middleware technology^[9]. CORBA and

DCOM are two most typical distributed middleware technologies. Nevertheless, CORBA or DCOM is a fairly complex, a task requiring special expertise. Quite often, the achievement of a good interoperability strategy is significantly constrained by many implementation restrictions in CORBA or DCOM^[9,23,24].

Web services have emerged as the next generation of integration technology^[7,23,25]. Based on open standards, the Web services technology allows any piece of software to communicate with each other in a standardised XML messaging systems^[6,9,23]. It solves and eliminates above issues of DCOM/CORBA^[23]. As a new type of software service, Web services are modular self-describing and self-contained applications that can be published, located and dynamically invoked across the Web. The Web-services technology is built on the foundation of open standards and common infrastructure. The Web-services framework is divided into three areas – communication protocols, service descriptions and service discovery, of which each is specified by an open standard^[26].

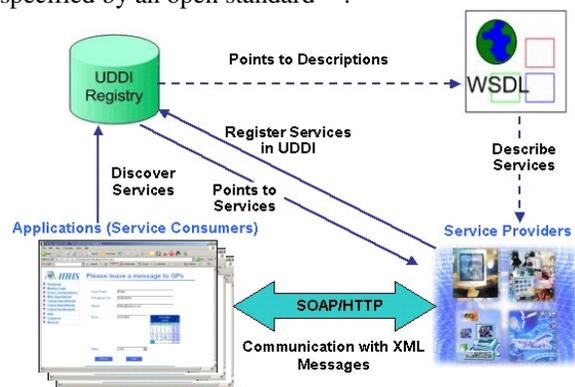


Fig. 1: A detailed architecture of Web service

Figure 1 indicates how web services works. In general, web services consist of two major technologies (XML – eXtensible Markup Language and SOAP – Simple Object Access Protocol) and two assistant technologies (WSDL – Web Services Description Language and UDDI – Universal Description, Discovery and Integration).

- * Firstly, service providers would make use of WSDL to describe their web services.
- Following the above step, service providers would register and publish their services in UDDI
- * Applications or service consumers find services via UDDI which would direct service consumers to relevant services according to the description of web services

* Regarding to previous step, applications or service consumers are able to invoke relevant web services using SOAP transmitted via HTTP on the Internet.

Web services encoded in XML, SOAP provides a way to communicate between applications developed with different programming languages and running on different operating systems. In fact, Web services provide a distributed computing technology for integrating applications on the Internet using open standards and XML encoding. The use of standard XML protocols makes Web services platform-, Language- and vendor-independent, thus an ideal solution for use in application integration. Figure 2 depicts how applications work with Web Services.

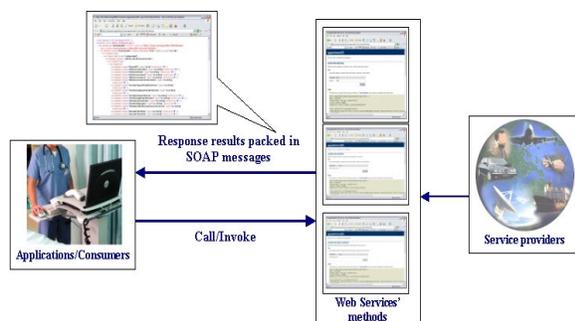


Fig. 1: Interaction between applications/consumers and Web services

With respect to Fig. 2, applications send requests and responses to and from Web services via SOAP. When a program invokes a Web-service method, the request and all relevant information are packed in a SOAP message and sent to the appropriate destination. When the Web service receives the SOAP message, it begins to process the contents (called the SOAP envelop), which specifies the method that the client wishes to execute and the arguments the client is passing to that method. After the Web service receives this request and parses it, the proper method is called with the specified arguments (if there are any) and the response is sent back to the client in another SOAP message. The client parses the response to retrieve the result of the method call.

Case study: Web Service-Based Integrated Healthcare Information System (WSIHIS):

A. Background of WSIHIS: Many people around the world became amputees because of war, traffic accident and some sorts of disease (e.g. Diabetes). Traditionally, these amputees would be treated with conventional socket techniques. With the development of medical technologies, new treatment's techniques for amputees

would be based on the Osseointegration that is discovered by Pro. Branenmark. The first Osseointegration surgery is dental implant and then he brought this technique further in 1990s^[27]. Pro. Branenmark applied the Osseointegration in orthopaedics. In 1997, Branenmark chose Queen Mary hospital as the clinical trail outside of Sweden^[27]. Basically, this technique makes use of Titanium implant as the attachment site for the artificial limb attachment.

Because this is entire new technique, patients would be selected carefully. A scheme called OPRA (Osseointegration Programme for Rehabilitation Amputee) was developed at Sweden that consists of patients' selection and recruitment, surgical plan after surgeon retirement and rehabilitation. The whole procedure of Osseointegration from the operation to the rehabilitation of patients lasts years. Overall, this process would involve doctors, patients, surgeons, prosthetic clinician and rehabilitation clinician. All relevant information must be recorded and documented for progress review. Patients' files are also required if the infection is developed in later stage. They would require an information system to manage and process these massive health data, as the patient, the hospital and the rehabilitation centre and prosthesis are normally not in same location. The data exchange between them would be very important. Consequently, WSIHIS is proposed to computerise all documents and data and offer a secure and stable environment for the communication between doctors and patients across the Internet.

B. Interoperability Issue of WSIHIS: According to the above statement, it is not difficult to find out WSIHIS would also face the challenging of interoperability issue as other HISs. Principally, interoperability issue of WSIHIS could be concluded as follows:

* **System interoperability issue:** Regarding to the description of background, WSIHIS would be required to link with various medical-relevant people (e.g. GP, Surgeon, prosthetic clinician and rehabilitation clinician), who might be using different system on their desktops to access to WSIHIS for different purposes. To achieve this feature, WSIHIS would be required to have the capability to interoperate with those different systems (e.g. Microsoft Windows Systems, Linux Systems, etc). Additionally, some parts of patient records could be produced from some existing systems used by some medical staff (e.g. GP) or

even depend on those data from existing healthcare systems. And with the data sharing purpose, WSIHIS would need to interoperate with different DBMSs (e.g. Microsoft SQL server, Oracle, Microsoft Access).

- * **Language interoperability issue:** WSIHIS is also expected to be integrated with functions of existing and future healthcare systems. Generally, different systems are built up in different technologies, particularly, different programming languages. For example, there are dozens of different patient-record systems in use across the NHS developed by couples of software providers. In order to enable WSIHIS to be compliant with those functions from different systems, WSIHIS is supposed to interoperate with applications built up in different programming languages, such as, Java, Visual Basic, C++ and C#.

C. Web Service-Based Solution for Interoperability Issue of WSIHIS: WSIHIS is built up with the support of Web Service to address above interoperability issue. Currently, there are two major application platforms – Microsoft .Net and Java 2 Enterprise Edition (J2EE) being able to create Web services. Microsoft .Net is selected as the major platform for developing the proposed system since Microsoft .Net would be more suitable than J2EE in terms of some features (e.g. overall maturity, interoperability, scalability, cost and the NHS perspective)^[5,28,29]. Besides, there have also been couples of developers of Healthcare Information Systems expressing the interest to Microsoft .Net for the development of their HIS^[2,5] when Microsoft .Net was not really put into the market. As the result, Microsoft .Net is selected to build up the proposed system. Figure 3 would illustrate a general concept of Web service-based solution of WSIHIS:

Within WSIHIS, Web service plays as the role of middleware that hides all these differences in system platforms, programming languages and database systems to users and developers. Accordingly, from users' perspective, they would be able to get the access to WSIHIS regardless of their different system platforms. From developers' perspective, they can invoke or reuse applications of WSIHIS in their systems with the support of web services.

Figure 4 explains how Web Service and Microsoft .Net technology are applied in WSIHIS. WSIHIS is built upon Microsoft .Net platform and Web service plays a role of middleware. Web services can be exposed from and consumed by any platform that can

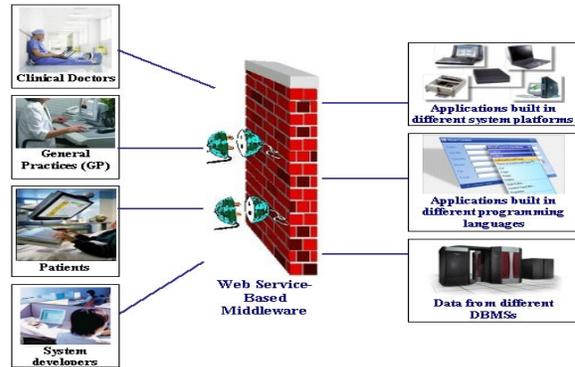


Fig. 2: The general concept of Web service-based solution for the interoperability issue of WSIHIS

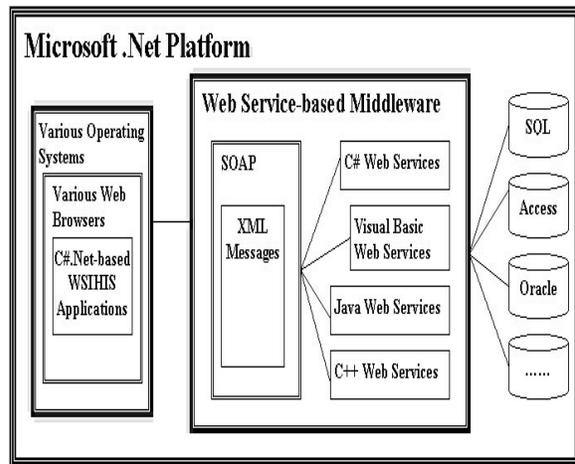


Fig. 3: The way of Web service working in WSIHIS with the support of Microsoft .Net technologies

format and parse an XML message because using XML for the formatting of requests and responses. This allows XML-based Web services to bring together disparate pieces of functionality – existing or new, internal or external to an organisation – into a coherent whole. Core technologies of Web service are XML and SOAP. Once Web services receive requests from applications, web services would retrieve data from different DBMSs (e.g. SQL server, Microsoft Access, Oracle, etc.) into datasets according to requirements of applications. All datasets would be written in XML messages, in additions, SOAP would act as an XML envelop to wrap those XML-based datasets into SOAP messages. And then these SOAP messages would be transmitted back to applications via HTTP.

Table 1: The methodology of whole system assessment

Type of the assessment	Methodology	Based on
Language Interoperability	Managing to integrate WSIHIS with applications built in different programming languages	Java, Visual Basic, C++, C#
System Interoperability	Trying to run WSIHIS on different platforms of web browsers	Internet explorer, Netscape, Mozilla Firefox, Deepnet, Avant browser, Opera
	Testifying if WSIHIS is able to work with different DBMSs	Microsoft SQL server 2000, Microsoft Access 2000, Oracle
	Trying to run WSIHIS on different platforms of operating systems	Microsoft Windows 2000, Microsoft Windows XP, Ubuntu Linux system 4.02, Knoppix Linux system 5.1.0

SOAP in Web service-based middleware provides a way to communicate between applications developed with different programming languages and running on different operating systems.

Additionally, Microsoft .Net platform is another important part of solution for the interoperability in WSIHIS. It is language neutral. It is best thought of as an open programming platform into which a variety of languages can be plugged. It is achieved by translating all different programming languages into a common language called Intermediary Language (IL). Firstly, source code is translated into Microsoft Intermediate Language (MSIL). This IL code is language-neutral and is analogous to Java bytecode. The IL code then needs to be interpreted and translated into a native executable. The .NET Framework includes the Common Language Runtime (CLR), analogous to the Java Runtime Environment (JRE), which achieves this goal. The CLR is Microsoft's intermediary between .NET developers' source code and the underlying hardware and all .NET code ultimately runs within the CLR. Regarding to the announcement of Microsoft company^[23], Microsoft .Net platform would also enable the system running under different system platforms and web browser platforms.

D. System Assessment on the interoperability of WSIHIS: The assessment has been done to testify whether Web service and Microsoft .Net technologies could really enable WSIHIS to overcome or eliminate the interoperability issue. The following part would make an analysis and discussion based on the result of assessment. The methodology used for system assessment is shown in Table 1.

Regarding to the assessment of language interoperability, four comparable Web service-based applications were built up in different programming languages (C#, Visual Basic, Java, C++). The purpose of assessment is to integrate four applications with

WSIHIS to achieve given tasks. The C# application is aiming to validate the reference number entered by the user. With this application, WSIHIS would display the error message for any invalid reference numbers entered by users.

In WSIHIS, there is a function named as 'Check Appointments' aiming to help doctors search appointments' records made by patients via reference numbers. Nevertheless, if the doctor forgets the reference number of patient, he would be unable to find the appointment record. Regarding to this reason, another function coded in Visual Basic was developed to enable the system display all appointments' records. Another application is developed in Java that is aiming to assist WSIHIS to find a relevant patient record via a reference number. The function of application in C++ is same as the application in Java that is to retrieve patient records through reference numbers.

The assessment of system interoperability comprises of testing the system on different web browsers, different DBMSs and different operating systems. Firstly, this assessment focuses on testing the interoperability of WSIHIS with six selected web browsers (Internet Explorer, Opera, Mozilla FireFox, Deepnet Browser, Avant Browser and NetScape), on which the author runs WSIHIS to execute specific tasks (e.g. making appointments, changing appointments, etc.) to see if WSIHIS is able to run regardless of different web browsers, of which the result has proved the proposed system has no problems on this point.

Secondly, the assessment is to testify the ability of WSIHIS for integrating different applications linked with three different DBMSs (Microsoft Access, Microsoft SQL Server, Oracle) and the result of this assessment verified that the system is able to interoperate with applications based on different DBMSs.

Finally, the assessment emphasises on testing the capability of WSIHIS running on different operating systems (Microsoft operating systems and Linux operating systems). WSIHIS was given specific tasks to carry out (e.g. making appointments, searching patients' records). The result of the assessment proved that the proposed system has no problem to work on any different operating systems.

DISCUSSION AND CONCLUSION

The intention of nationalising and even globalising healthcare systems by using computer technologies has become a very common talking point. Since at the end of 2003, British government started to develop a

national –scale computerised healthcare system, named as National Program for IT (NpfIT). NpfIT is aiming to integrate most of healthcare systems and services based in England and Wales. However, this project was severely delayed for years because of various issues, such as data transmission, network limitation, data security, short of the cooperation of medical staff and so on. Among these issues, the issue of system interoperability is becoming more critical since there are considerable hospitals in England and Wales that running in different business logic. And lots of hospitals have built up their own HISs and database systems. This situation may cause one patient having multi medical records in different hospitals or even in different departments in one hospital. In addition, those HISs in hospitals could be developed different computer languages, compiled on different platforms, executed on different hardware, hand have different data structures, types and formats. These systems were not developed in a coordinate way but evolved as autonomous and heterogeneous systems. Thus, in most of the cases these information systems function independently and do not share their data and process. Computer technologies are keep developing and changing. If a system is poor in its interoperability, any increasing functions or little changes could stop it working properly. Hence, the issue of system interoperability is becoming critical for NpfIT or any big scale HISs to overcome. Regarding to this issue, some developers are trying to make use of a traditional distributed middleware technology (DCOM/CORBA) to solve it. However, DCOM/CORBA is relatively complex to handle and requires some special expertise. Any large-scale systems have been very difficult to develop and maintain, as the result, they should be technically easy for the development and maintenance. So that developers are looking for another more suitable solution. In this case, a new Healthcare Information System is proposed and implemented in this thesis in order to address the interoperability issue, named as Web Service-based Integrated Healthcare Information System (WSIHIS).

A new distributed middleware technology, Web Services, plays a key role in WSIHIS. Based on open standards, the Web services technology allows any piece of software to communicate with each other in a standardised XML messaging systems^[8,19,12]. It solves and eliminates above issues of DCOM/CORBA^[12,20]. As a new type of software service, Web services are modular self-describing and self-contained applications that can be published, located and dynamically invoked across the Web. The core technology of Web Services

is Simple Object Access Protocol (SOAP) and Extensible Markup Language (XML). SOAP is a platform-independent protocol that uses XML to make remote-procedure calls over HTTP. SOAP message is written in the easy-to-understand and platform-independent XML. HTTP was chosen to transmit SOAP, because HTTP is a standard protocol for sending information across the Internet. The use of XML and HTTP enables different operating systems to send and receive SOAP message. Web Services allow client and server implementations to construct their distinct but equivalent representations of any data structures. Web services that use SOAP support a wider variety of data types. The data types supported by SOAP include most basic data types, as well as Dataset, DateTime, XmlNode and several others. SOAP also permits the transmission of arrays of all these types.

Compared to CORBA/DCOM, the technology of Web Services could reduce development and deployment time, make the system implemented in the low cost, reduce the maintenance complexity and lower the risk of project failure. Since any big projects like NpfIT are required to be easy for the maintenance, short development period, low implementation cost and low risk of project failure, these features make Web Services suitable for any large-scale HISs.

According to the standard system development lifecycle, a model of WSIHIS was designed and implemented based on Microsoft .Net platform by using C# programming language. In order to prove if Web Services is a right solution for the interoperability solution, an assessment has been done concentrating on the system interoperability from several aspects. The major purpose of assessment tries to make sure if WSIHIS could be run on different system platforms (e.g. Linux system platforms, Windows System Platforms, etc.), integrated with some applications developed in different computer programming languages (e.g. Visual Basic, C++, Java and C#) and different database systems (e.g. Oracle, Microsoft Access, SQL server). The result drawn from the assessment is satisfied. As the result, the author has reasons to believe Web Service-based solution based on Microsoft .Net platform by using C# programming language would be suitable for overcoming the issue of system interoperability for current healthcare information systems.

Web Service supports increased interoperability but also represents a significant increase in run-time cost for Web Service solutions. Conversion to text format and parsing of XML documents is inherently more costly than the alternative mechanisms used to

convert data to a common data representation for the network. The additional communications and processing costs are frequently perceived as a potential barrier to the use of Web Services technologies. Hence, authors will try to find out a solution to overcome these shortcomings of Web Services in future.

REFERENCES

1. DH and NHS, 2003. Connecting For Health - A Public-Private Collaborative. Connecting For Health, NHS: England.
2. Matsopoulos, G.K., V. Kouloulis and P. Asvestas, 2004. MITIS: a WWW-based medical system for managing and processing gynecological - obstetrical -radiological data. *Computer Methods and Programs in Biomedicine*, 76: 53-71.
3. Tsiknakis, M., D.G. Katehakis and S.C. Orphanoudakis, 2002. An open, component-based information infrastructure for integrated health information networks. *Intl. J. Med. Informatics*, 68: 3-26.
4. Wreder, K. and Y. Deng, 1999. Architecture-centered enterprise system development and integration based on distributed object technology standard. *Computer Software and Application Conf.*
5. Varge, B. and P. Ray, 2003. Interoperability of hospital information systems: A case study. *Enterprise Networking and Computing in Healthcare Industry*, 5th International workshop.
6. Deitel, H.M., P.J. Deitel, J. Listfield and T.R. Nieto, 2002. *C# How to program - introducing.net and web service*. London: Prentic-Hall, Inc.
7. Chester, T.M., 2005. Cross-platform integration with XML and SOAP. *IT Professional*, 9: 67-70.
8. Duthie, G.A., 2003. *Microsoft ASP.Net programming with microsoft visual C#.Net step by step*. Washington, USA: Microsoft Press.
9. Zhu, J., 2003. Web services provide the power to integrate. *Power and Energy Mag.*, 1: 40-49.
10. Pronab, G. and R. Pradeep, 2000. *Software Interoperability of Telemedicine Systems: A CSCW Perspective*. IEEE.
11. Maglaveras, N. and I. Chouvarda, 2003. The citizen health system (CHS): A modular medical contact center providing quality telemedicine services. *IEEE Trans. Inform. Technol. In Biomed.*, 9: 3.
12. Jablonski, S., R. Lay and C. Meiler, 2005. Data logistics as means of integration in healthcare applications. *ACM Symp. Appl. Comput.*, 1 - 58113-964.
13. NpFIT, 2005. *Making IT happen - Information about the National Programme for IT*. Department of Health: England.
14. James, A.E. and Y.H. Wilcox, 2001. A telematic system for oncology based electronic health patient records. *Inform. Technol. Biomed.*, 5: 16-17.
15. Cross, M., 2004. In *Sickness or In Health?* IEE Review, 50: 10.
16. Albahari, B., P. Drayton and B. Merrill, 2001. *C# Essential - A Comparative Overview of C#*. 1st Edn. (Ed. O'Reilly).
17. NHS, 2005. *History of Connecting for Health*. NHS Connecting for Health, Department of NHS: England.
18. Guah, M.W.C., 2003. *Logicity of ASP in healthcare: The NHS case study*. Proc. 37th Ann. Hawaii Intl. Conf., Hawaii: IEEE.
19. NHS, 2005. *Electronic Transmission of Prescriptions (ETP) Programme*. NHS Connecting for Health: England.
20. DH and NpFIT, 2005. *Choose and Book Service Implementation Guide*. NHS Connecting for Health: England.
21. Mathews, A.G. and R. Butler, 2005. A vision for the use of proactive mobile computing tools to empower people with chronic conditions. Proc. 18th IEEE Symp. *Computer-Based Medical Systems (CBMS'05)*, IEEE.
22. DH and NHS, 2005. *NHS connecting for health fact sheet*. Connecting for Health, NHS: England.
23. Banerjee, A., A. Corera, 2002. *C# Web Services - Building Web Services With.NET Remoting and ASP.NET*. WROX.
24. Gokhale, A., Kumar, B., Sahuguet, A. *Reinventing the Wheel? CORBA vs. Web Services*. in The Eleventh International World Wide Web Conference. 2002. Honolulu, Hawaii, USA.
25. Umar, A., 2004. The emerging role of the web for enterprise applications and ASPs. Proc. IEEE.
26. Curbera, F., M. Duftler and R. Khalaf, 2002. *Unraveling the Web Services Web - An Introduction to SOAP, WSDL and UDDI*. IEEE Internet Computing.
27. Sullivan, J., M. Uden and P. Robinson, 2003. *Rehabilitation of the trans-femoral amputee with an osseointegrated prosthesis: The United Kingdom experience*. Prosthetics and Orthotics International.
28. Sessions, R., 2001. *Java 2 Enterprise Edition versus The.Net Platform Two Visions for eBusiness*, ObjectWatch, Inc.: Texas.
29. Vawter, C. and E. Roman, 2001. *J2EE vs. Microsoft.Net - A comparison of building XML-based web services*, The Middleware Company.