PROCEEDINGS OF THE 13th ANNUAL CONFERENCE ON WORLD WIDE WEB APPLICATIONS

14-16 September 2011
Johannesburg
South Africa

Editors:

A. Koch
P.A. van Brakel

Publisher:

Cape Peninsula University of Technology
PO Box 652
Cape Town
8000

Proceedings published at
http://www.zaw3.co.za

ISBN: 978-0-620-51918-2

TO WHOM IT MAY CONCERN

The full papers were refereed by a double-blind reviewing process according to South Africa's Department of Higher Education and Training (DHET) refereeing standards. Before accepting a paper, authors were to include the corrections as stated by the peer-reviewers. Of the 59 full papers received, 41 were accepted for the Proceedings (acceptance rate: 69.5%).

Papers were reviewed according to the following criteria:

- Relevancy of the paper to Web-based applications
- Explanation of the research problem & investigative questions
- Quality of the literature analysis
- Appropriateness of the research method(s)
- Adequacy of the evidence (findings) presented in the paper
- Technical (e.g. language editing; reference style).

The following reviewers took part in the process of evaluating the full papers of the 13th Annual Conference on World Wide Web Applications:

Prof S Berman
Department of Computer Science
University of Cape Town
Cape Town

Prof RA Botha
Department of Business Informatics
Nelson Mandela Metropolitan University
Port Elizabeth

Mr AA Buitendag
Department of Business Informatics
Tshwane University of Technology
Pretoria

Prof AJ Bytheway
Faculty of Business
Cape Peninsula University of Technology
Cape Town

Dr A Chigona
E-Learning Support and Innovation Unit
University of the Witwatersrand
Johannesburg

Prof T du Plessis
Department of Information and Knowledge Management
University of Johannesburg
Johannesburg

ISBN: 978-0-620-51918-2

Dr L Harrison
Educational Technology, CELT
Durban University of Technology
Durban

Prof M Herselman
Meraka Institute, CSIR
Pretoria

Mr B Kalema
Department of Business Informatics
Tshwane University of Technology
Pretoria

Ms F Mohsam
Faculty of Business
Cape Peninsula University of Technology
Cape Town

Dr J Mostert
Centre for Development Support
University of the Free State
Bloemfontein

Prof L Nagel
Department of Education Innovation
University of South Africa
Pretoria

Ms C Muir
Department of Strategic Communication
University of Johannesburg
Johannesburg

Mr R Proske
University Library
Cape Peninsula University of Technology
Cape Town

Mr F Schwenke
Faculty of Informatics and Design
Cape Peninsula University of Technology
Cape Town

Prof A Singh
Business School
University of KwaZulu-Natal
Durban

Prof JS van der Walt
Department of Business Informatics
Tshwane University of Technology
Pretoria

Prof D van Greunen
School of ICT
Nelson Mandela Metropolitan University
Port Elizabeth

Dr SC Warden
Faculty of Informatics and Design
Cape Peninsula University of Technology
Cape Town

Further enquiries:

Prof PA van Brakel
Conference Chair: Annual Conference on WWW Applications
Cape Town
+27 21 469 1015 (landline)
+27 82 966 0789 (mobile)

ISBN: 978-0-620-51918-2

Design of an integrated real-time information sharing model for a train incident response and handling between related departments at WitsMetrorail

B.N. Gatsheni
University of Johannesburg
Johannesburg, South Africa
bgatsheni@uj.ac.za

Ike Sikakane
ike@vut.ac.za

J. Mangane
jethrom@vut.ac.za

Vaal University of Technology
Vanderbijlpark, South Africa

Abstract

The average time (response time) it takes currently for the information on a train that has stopped in the middle of nowhere to flow from the train crew through related departments (operations, planning, maintenance) at Wits Metrorail headquarters before technicians can leave for the place of Incident is 40 minutes. This time is too long and by the time the technicians get to the train passengers might be frustrated to the point of torching the train coaches. This has resulted in a loss of assets. The UML's Sequence diagram has been used in this work because of the following features: it models real-time applications and shows explicit sequence of stimuli messages exchanged between object instances participating in the interaction. Thus its use by related departments at Metrorail ensures fast exchange of train Incident information within these departments. SOAP, web services and mobile computing applications form part of the model. An integrated real-time information sharing model that enables related departments to see information at the same time has been developed. This model can reduce the current response time to train Incident by 75%. The model ensures accuracy and reliability of information through the communication chain and also fast exchange of information between related departments than any system that we know of within the railway companies in South Africa.

Key words: Web services, semantics, soap, UDDI, fast exchange of information, WSDL

Introduction

Wits Metrorail provides train commuter services to about 1.4 million commuters daily in the Johannesburg metropolitan area. Train delays are common due to cable theft

and accidents among others. A delay in arrival by technicians can frustrate commuters who tend to resort to burning trains and hence resulting in a loss of assets.
The focus of this work is on a train that has stopped in the middle of nowhere. Central to resolving this problem is the related departments also to be called train Incident response and handling departments in this paper. These departments include operations, maintenance, planning and spares and procurement.
Currently, when there is an Incident, the train crew signals the train operations center (monitors and manages trains) who in turn alerts the planning department to prepare a job card and issue it to the technicians. Technicians pass through the planning department collecting the job card and then pass through the spares department collecting spares en-route to the scene of the Incident. This last process can take over 10 minutes. The report from the train crew is repeated by each department on the communications chain until it gets to the end in Figure 1. If it took the crew 5 minutes to explain to the train operation centre, it will take 5 minutes for each department to explain to the next and by the time it gets to the last department, 20 minutes will have elapsed and the accuracy and reliability of information might be compromised. This is not acceptable.

The current Incident response and Incident handling model at Wits Metrorail

Figure 1 is a sequence diagram showing the current incident reporting procedure used at Wits Metrorail.

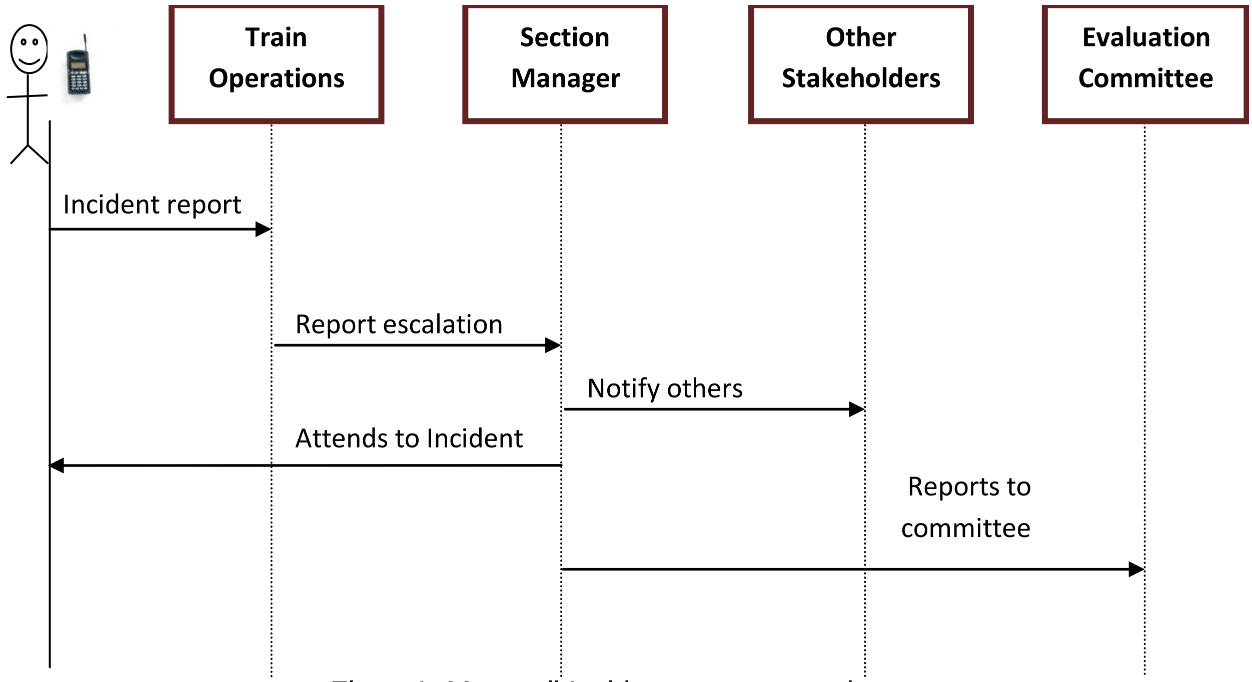

Figure1: Metrorail Incident report procedure.

In Figure 1, a train driver or crew member will send sms (or use a truncin device) to train operations center (TOC), notifying them of an incident, A site coordinator at

Proceedings of the 13th Annual Conference on World Wide Web Applications
Johannesburg, 14-16 September 2011 (http://www.zaw3.co.za)
ISBN: 978-0-620-51918-2

TOC will then relay message to a section manager who will in turn classify the incident (either as Yellow, Orange or Red). Classification of the incident, determines how serious the incident is and the priority it should receive. The section manager then goes to the scene to normalize the situation and liaise with relevant departments. After the situation has been normalized, the section manager will then send a report to the board of enquiries or evaluation committee which will give recommendations on actions to be taken.

Limitations of the current Incident response procedure:

- The current system is not automated (manually operated) and is subject to a lot of human errors.
- Information is not integrated as it largely depends on site coordinators to relay incident information to related departments. This can be problematic and further delays the response to train incidents.

Figure 2 shows the proposed train Incident response and handling model that is based on the unified modeling language (UML) sequence diagram.

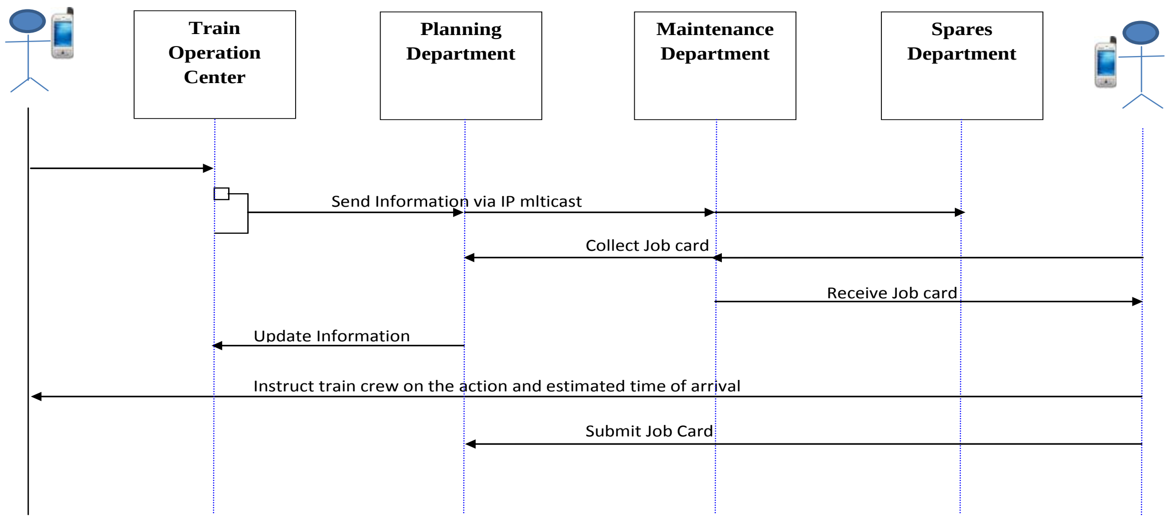

Figure 2: UML Sequence Diagram for Information Exchange between Incidence response and handling departments at Metrorail

UML provides specification, visualization, construction and documentation of software. UML's Sequence diagram in Figure 2 shows the proposed pattern of interaction among related departments. It models real-time applications which ensure the exchange of information by related departments takes place in real-time. Figure 2 shows the train crew interacting on train Incidents with the system and the sequence of messages between interacting related departments.

ISBN: 978-0-620-51918-2

The attributes of the Sequence diagrams which are relevant for the fast exchange of information between the related departments at metro rail are as follows:

- Its temporal dimension (order of messages in predefined time slots) and facilitation for concurrent processing.
- it depicts the order of events or pauses between events better than the collaboration diagram, a key attribute when modeling the fast information exchange between related departments (a real-time system).
- it can model a “race condition” e.g. 2 events interrupting precisely at the same time which is a likely scenario with interaction between related departments.

In Figure 2 a smart phone combines the functions of personal digital assistants (PDA) and a Cellular phone. Relevant PDA features for the model include providing internet capabilities, mobile computing and information retrieval capabilities through Universal Description, Discovery and Integration (UDDI). A PDA provides speed and security through the virtual private networks (VPN) for transmitted data from the remote train crew at the scene of the Incident to the train operation center. In addition, a PDA:

- allows technicians to use twitter to share information when troubleshooting

Another component of the model for exchanging information is the simple object access protocol (SOAP) and web services.

SOAP in the fast exchange of information

SOAP in this model is used to access web services. It is a protocol based on HTTP and eXtensible Markup Language (XML) and is used as a format for sending messages between applications or related departments in a distributed computing environment (related departments). The HTTP protocol is popular for use with SOAP because of HTTP’s (is supported by all browsers and servers) wide use on the Internet and that HTTP can bypass firewalls and proxy policies. The exchange of information of SOAP over HTTP involves a request and response strategy.
SOAP’s rivals are DCOM and CORBA and they are not suitable for our model as they communicate using RPC between objects. RPC suffers from compatibility and security problems. Firewalls and proxy servers normally block RPC traffic.

Web services in Incident response and handling

The webservice in the model forms part of the whole infrastructure that includes all parties involved in Incident response and Incident handling. Web Services use SOAP for message exchange between a service client and a service provider. A Web service is a web API or software module components that are wrapped inside a set of protocols (such as SOAP and WSDL) and in our model are delivered over the intranet via XML messaging and communicate with other components automatically without human intervention. Web services can be accessed over a network and executed on a remote system (train operation centre) that is hosting the requested service (by train crew). Related departments, the train operation centre, the train crew PDAs and also technician PDAs all act as clients and servers which on a web service communicate over HTTP.

What makes web services attractive for the model is that they enable related departments to exchange data between disparate software applications (e.g. word processors) written in different programming languages [6] running on a variety of platforms (e.g. Linux) in a secure way (a great saving in terms of needing new skills and also new software).

The need for program-to-program communications necessitates that Web services be described in detail so other web services can discover it (using the Universal Description, Discovery and Integration (UDDI)) and know how to connect to it. Our model exploits in addition to SOAP, the following elements shown in Figure 3:

- WSDL (web services description language) - an XML based language used for locating and describing the web service.
- UDDI - a web service directory equivalent of yellow pages stores information about web service location and enables this webservice's discovery by other web services and transport protocols [1,2] and it communicates via the SOAP protocol.

In this model Wits metro rail will register its web services in a global directory. Thus applications on the crew's PDA with client and server software and UDDI are able to dynamically discover new web services automatically, e.g. access history cards in the planning department or new tools at spares department.

A web service and its relevant building blocks

An Agent in Figure 4 is a piece of software or hardware that sends and receives messages. Semantics describes rules with regard to the behavior of the service in response to messages sent between the requester and provider in Figure 4. They are seen as a contract between these parties regarding the scope of interaction.

Our incorporation of a Service Oriented Architecture (SOA) was informed by SOA's find-bind-execute approach. With SOA related departments register their services in a private (enterprise) registry, which clients use to locate Web services. If a service registry has information on a service request by a client, it provides a contract and an access point.

ISBN: 978-0-620-51918-2

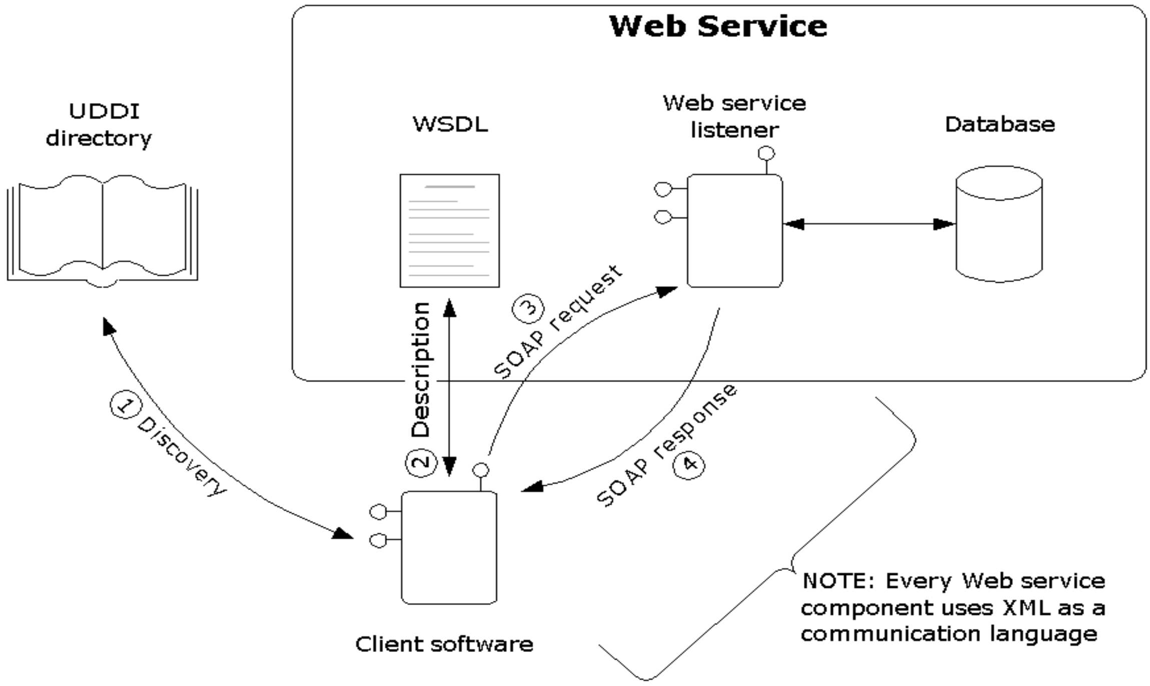

Figure 3: This is a web service used for the train Incident response and handling

In Figure 3: Database – the planning department will request for spares by submitting job card details to the stores database. By the time technicians get to stores, spares will be ready for them.

The requester and the provider shown in Figure 4 from web services use the train operation centre (TOC) and Incident response departments. The Requester and Provider take this form:

- related department(s) become known to each other (establish a connection).
- agents exchange messages in Figure 4, thus performing some tasks on behalf of the requester (train crew or technicians) and provider (train operation centre and Incident handling departments in Figure 3).
- entities broker an agreement between them shown in Figure 4 point 3 on WSD and Semantics (expectations of requester and provider regarding information sharing) that will govern the interaction as shown in Figure 2 and in Figure 4.

SOAP offers information exchange but does not tell what messages must be exchanged to interact with a service. That role is done by WSDL, an XML format which describes Web services as communication end points that exchange messages, Web service’s interface and provides users with a point of contact. In Figure 2, PDAs of technicians and each department have client and server software to afford 2 way communication.

ISBN: 978-0-620-51918-2

Requester Entity (Incident response departments at Wits metro rail) **Provider Entity (Wits Metro rail Operations Center)**

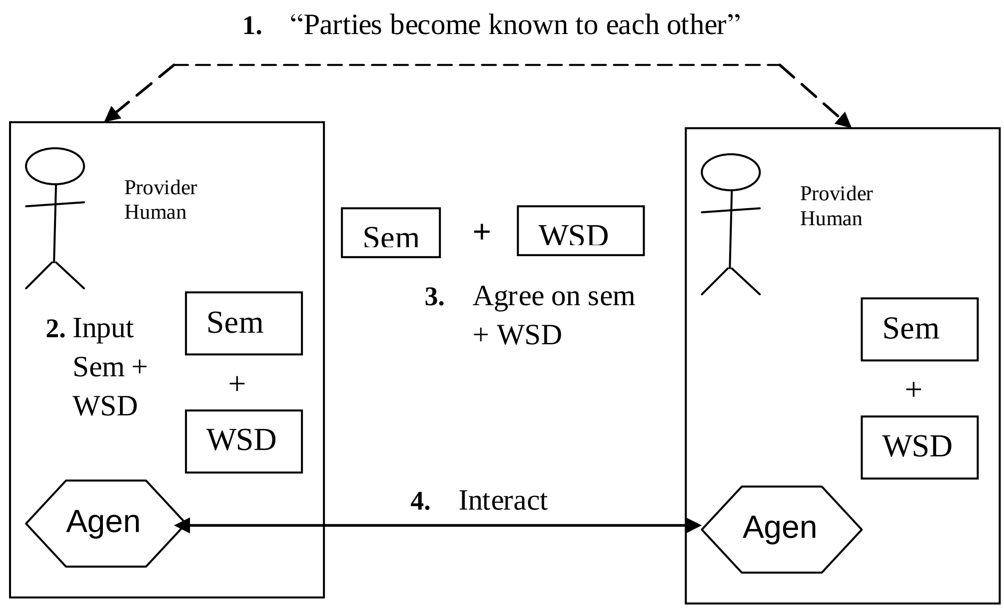

Figure 4: An illustration of a web service

Adding semantics to Web Service description for the model will enable the automation of discovery, composition and invocation of Web services.

Choreography and orchestration in train Incident handling

Choreography [2], a web service component provides the specifications on how messages should flow between diverse, interconnected components and applications to ensure optimum interoperability between interacting applications. It can consist of a series of orchestrations [2]. Its goal is to describe collaboration between a collection of services to achieve a common goal. This collaboration and interaction is between the Incident response and handling departments whose goal is a quick response to train Incidents. Orchestration [2] is the ability to control information flows and how web services interact to accomplish tasks between many systems. This interaction is within and between departments and data that binds them together.

Multicasting in train Incident handling

Multicast in Figure 1 seamlessly deliver train incident data to related departments at the same time. IP multicasting sends a packet only once to multiple recipients on a network simultaneously. The competing protocols for IP multicast include UDP and pragmatic general multicast (PGM). UDP, a connectionless internet protocol means packets may arrive out-of-order, duplicated or missing without notice. In time-sensitive applications UDP is key as dropping packets are preferred to waiting for delayed packets. However, it will be a disaster if complete information does not reach the train operation centre.

ISBN: 978-0-620-51918-2

PGM provides a reliable sequence of packets to multiple recipients simultaneously [7]. PGM uses Negative acknowledgment (NAK) of packets in case of data loss or out-of-order packets sent to the receiver to retransmit data stream. Applications for PGM are in the stock market sticker update information and weather among others [7]. In this work PGM will be used. Figure 5 is incorporated in the real-time data sharing in Figure 1.

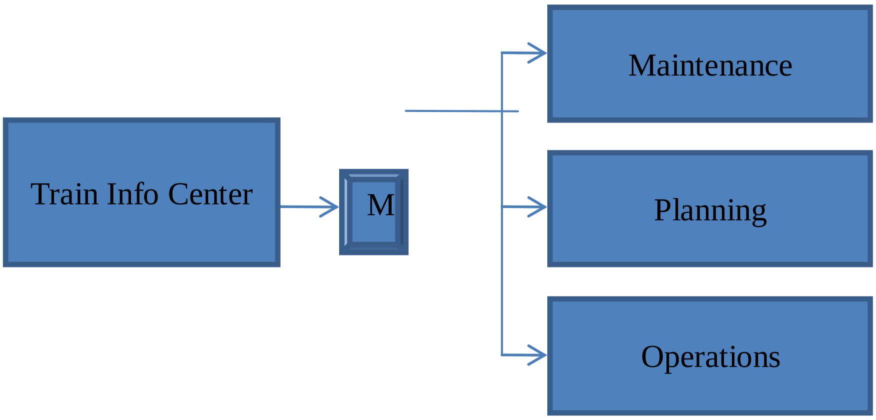

Figure 5: This is a multicast architecture based on PGM for Incident handling departments.

In Figure 5 when the message is received at the train operation centre, a multicast unit “M” broadcasts the message to the 3 departments (shown in Figure 1) which minimizes delays to react to the train Incident.

There is need to secure the information channel between the crew or technicians in a remote location and the local area network (LAN) of Wits Metro rail.

Virtual Private Networks (VPN)

VPN is a network that is implemented on top of an existing public network [8] to connect remote users to their company’s network. In Figure 1 VPN is implemented in a PDA. The aim is to access the Wits metro rail network securely. VPN in the model secures the communication link between the train crew and Wits Metrorail headquarters.

VPN uses tunneling protocols like Internet Protocol Security (IPsec) to provide confidentiality, sender authentication and content encryption to achieve privacy [8]. The key features of IPSec suite that are incorporated into the model are:

Internet Key Exchange (IKE) to setup a security association (SA) by handling negotiation of protocols and through Encapsulating Security Payload (ESP ensure data origin authentication, integrity and protection against replay attack).

Discussion

A real-time Incident response, handling and information sharing model for a train that has stopped in the middle of nowhere has been designed. It potentially improves the exchange of information between these departments by 75%. This information sharing model will enable these departments to see information on train Incident at the same time and thus facilitate a quick response to the Incident.

Using web services, the cost of bandwidth is still prohibitive in South Africa despite the advent of the undersea cable. However, the notion of different applications, written in different programming languages and running in different platforms but interoperating transparently is cost effective (no need for new software or acquisition of new skills) and supports interoperability between Incident response and handling departments. SOAP offers portability which is key for the train crew and also technicians.

There are some limitations in terms of the technology used in our model. Data is converted to a textual form before it can be transmitted in a SOAP message. Fortunately for the train crew and technicians, data involved is small. However, since SOAP communicates using XML which is textual and verbose and thus converting data to XML and back negatively affects application performance and hence is inefficient. Furthermore, when a couple of transactions request similar service operations (a likely scenario with Incident handling departments), SOAP unicast kicks in generating vast amounts of traffic. The remedy will be to reduce SOAP network traffic. The web services because they are a distributed application suffer from high latency and high protocol overhead.

Another limitation is the UDDI which does keyword search for service retrieval and cannot fully convey client service queries which makes it unreliable. Its rival, the search engines are not suitable for web services. Web pages contain plain text and web services have a small portion of plain text. Thus search engines can use information-retrieval such as word frequencies for the former but will fail to discover the latter. Finding patterns within web services is difficult as interface information such as parameter names can vary in a big way. The key features of the model are fast response time, low cost and high availability. Key is the extent to which the model can deliver the required functionality and meet requests. The scientific contribution of this paper is in terms of application of the technology to the commuter rail system.

Conclusion

An Incident response and handling model for a train that has stopped in the middle of nowhere has been designed. It is a real-time model that potentially improves the exchange of information between these departments by 75%. Now it takes 5 minutes explanation to all related departments simultaneously instead of the current 20 minutes.This real-time information sharing model will enable Incident response and handling departments to see information on train Incident at the same time (like a flow chart hanging from a wall) and thus facilitate a quick response to the Incident.
There is currently no model that we know of to date that has exploited this technology in this way for the fast exchange of information between related train Incident handling departments.

References

1. *Than DT, Hoang NH, Choi E. 2007. The WORKGLOW system in P2P based web service orchestration.* In: *IEEE International Conference on Convergence Information Technology,* November 2007. Gyeongju: 2313—2317

ISBN: 978-0-620-51918-2

2. *Suazo NC, Aguire JOO. 2005. Aspect-Oriented Web Service orchestration. In: 2nd International Conference on Electrical and Electronics Engineering (ICEEE)*; 7-9 September, 2005. Mexico City: 72 - 76

3. *Gu N; Cui J; Ye W; Wang H; Pei J. 2007. A System Framework for Web Service Semantic and Automation Orchestration. In: 2nd International Conference on Pervasive Computing and applications,* 26-27 July 2007. Birmingham: 606 - 611

4. *Ma, C, He, Y. 2009.* Chao Ma; Yanxiang, He. 2009. *An Approach for Visualization and Formalization of Web service Composition. In: International Conference on In: Web Information Systems and Mining*, 2009 : 342 – 346

5. *Muyle, R. Miller, J. Verma, K. Gomadam, K. Sheth, A.2005. A Semantic Template Based Designer for Web Processes. In: IEEE Proceedings on International Conference on Web Service,* 11-15 July 2005:

6. Dietze, S. Gugliotta, A.; Domingue, J. *Situation-Driven processes for Semantic Web Services. In: International Conference on Service Systems and Service Management*, June 30 -July 2 2008. Melbourne: 1 – 5.

7. Gemmell, J. Montgomery, T. Speakman, T. Crowcroft, J. The PGM reliable multicast protocol. In: *IEEE Network Magazine ; Volume: 17 Issue:1*

8. Zhang, Z. Gao, L. Xie, H. Tao, Q. Implementation of the load balancing for multiple VPN server. *In: International Conference on Educational and Network Technology (ICENT)*, 25-27 June 2010. China: Qinhuangdao: 147 – 150.

Acknowledgements:

1. Mr Nelson Mphailane of Prasa (formerly with Wits metro rail).
2. Motloli Letsae of Wits Metro rail.