Breaking Barrier to Technology: e-Governance Messaging Middleware

Vijay Jain, Rajeev Srivastava, Ranjan Kumar, Rahul Upadhyay, Kapil Kant Kamal
Centre for Development of Advanced Computing (C-DAC), Mumbai, India
{vijayj, rajeevs, ranjank, rahulup, kapil}@cdac.in

ABSTRACT
Service Oriented Architecture (SOA) has emerged as a successful technology paradigm for interconnecting applications in heterogeneous environments. National e-Governance Service Delivery Gateway (NSDG), a SOA based messaging middleware, routes messages across government departments thereby enabling cross state and cross domain service delivery, overcoming challenges of interoperability and integrations while delivering services to citizens through a single window. This paper discusses the relevance of standard based messaging middleware for integrating services at national, state and local government, and interoperability and testing challenges for such kind of implementation.

Categories and Subject Descriptors
D.4.4 [**Communications Management**]: *Message sending*; H.4.3 [**Communication Applications**]

General Terms
Design

Keywords
Interoperability, SOAP, SOA, Web Services, Messaging Middleware, e-Governance, Gateway

1. INTRODUCTION
Emergence of e-Governance has increased the adoption of information and communications technologies (ICTs) at national and international level. e-Governance has successfully transformed the traditional ways of delivering government services and is bridging the intercommunication gap between government and citizens. In India, e-Governance projects have been undertaken on a large scale to cater to citizen requirements and facilitate government services in an efficacious manner. Providing these joint services in a transparent, reliable and timely manner is an underlying operational goal along with addressing the challenges of integrating various services that are developed on heterogeneous platforms. The National e-Governance Service Delivery Gateway (NSDG) [14], a Mission Mode Project (MMP) under National e-Governance Plan (NeGP) [15], have been implemented by Government of India (GoI) to achieve the goal of seamlessly integrating applications across government departments. NSDG provides capability to route messages across departments, thereby enabling cross state, cross domain service delivery, overcoming challenges of interoperability and integration while delivering services to citizens through a single window. This paper discusses the relevance of NSDG, a standard based messaging middleware for integrating services at national, state and local government. Interoperability and testing challenges are also being discussed. Hereafter NSDG will be referred as Gateway in this paper. The remaining part of the paper is organized as follows. Section 2 summarizes the implementation of e-Governance in other countries. Section 3 discusses middleware requirements and its applicability in India. Section 4 elaborates the structure of Gateway middleware. Section 5 focuses on challenges faced while integrating and testing the Gateway. The paper concludes with specific remarks and future work.

ICEGOV2011, September 26–28, 2011, Tallinn, Estonia

2. RELATED STUDIES
ICT brings the efficiency and effectiveness to citizen services and transactional exchanges within government (G2G), citizens (G2C) and businesses (G2B) [6]. Stakeholders which have been normally identified within e-Government are government, citizens and business [5] [11]. e-Governance system should provide a standardized platform to citizens to provide services through its legacy systems and the information should be accessible through various channels also [13]. Researchers have explored the importance of e-Government interoperability; e.g. Cava & Guijarro [3], Benamou et al [2], Klischewski [8], Bekkers [1], Klischewski & Scholl [9]. Few of them defined organizational, legal and social aspects more challenging than that of technical interoperability [12]. European Interoperability Framework (EIF) has been discussed proposing three dimensional view of interoperability which are technical, semantically and organizational [4]. Factors influencing the architecture for building e-Government interoperability including the challenges of semantic and pragmatic interoperability are being discussed [7]. Authors [10] suggest grid system based on multi-agent and use Government Information Metadata Registry (GMDR) as a basic structure for supporting interoperation of legacy systems. In this paper we have shown syntactic and semantic interoperability through Gateway middleware for diversely spread services at multiple levels of the Government, whereas other aspects of interoperability are also in consideration as future work.

3. MIDDLEWARE REQUIREMENTS
GoI aims to provide its service accessibility to common man through various service delivery channels accomplishing the purpose of efficient service delivery at affordable cost. Government systems are built on heterogeneous platforms, technologies and are spread across diverse geographical locations, in varying state of automation, making this task very challenging. One of the major initiatives of Indian Government is to integrate information and improve collaboration among government departments at local, state and national level. Middleware in the form of Gateway is developed to achieve the goal of common platform for service delivery. Gateway reduces the efforts required to integrate applications built on different technologies and platforms, facilitating speed and effectiveness of information exchange. Government now has unprecedented flexibility in

integrating its services across the country using the capability of Gateway which is implemented in SOA.

Gateway as Middleware

Gateway is built on standards named as eGov exchange [16], defined by GoI for the purpose of message communication between government departments. These standards conform to the standards defined by W3C, XML, WS-I and SOAP for message communication purpose and provides underlying architecture to resolve the interoperability, integration, scalability and reliability problems at large scale. The future e-Governance space of India would see government departments/local bodies and businesses offering many services which will be consumed by the citizens, businesses and other government departments/local bodies. This would give rise to multiple Service Access Providers (SAP) and Service Providers (SP) and these necessarily may not reside with one Gateway but may be distributed among more than one Gateway (Gateway Constellation). The choice of state Gateway will be dictated based on the need of a state for connecting e-Governance services on multiple technologies/platforms to increase interoperability factor. In absence of a Gateway in any vertical or state level, services can still be offered through national Gateway. For example state police service can have access to national passport service, either through national Gateway or state Gateway. If the service is associated with state Gateway then it can be reached through constellation of Gateways, otherwise it can be directly accessed through national Gateway.

Gateway acts as an intelligent hub and routes service requests from service seeker, commonly referred as Service Access Provider (SAP) to Service Providers (SP). SAP provides front end access medium for citizens and businesses to avail e-Governance services. SPs are geographically distributed back-end departments offering e-services to citizens and businesses, and to other government departments. SAP is always in need of accessing SPs which are providing their services for electronic delivery. Before the Gateway, SAP had to integrate with every SP separately, creating the mesh architecture as shown in Figure 1.

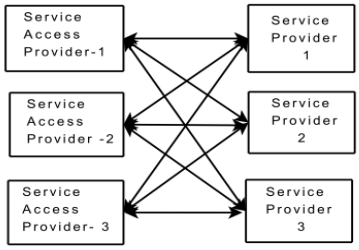

Figure 1. Mesh Architecture of SAP and SP

This mesh architecture resulted in tight coupling of the systems involved. In this integration even the communication between every SAP and SP may differ in technology frameworks and standards for their applications or services. Communication protocols may also vary between different SAPs and SPs. Single SP can offer multiple services associated with it. There is a possibility of diverse applications developed on varying platforms, integrated to the Gateway reducing the mesh as shown in Figure 2.

Integration is possible through eGov exchange specifications [16] for standardizing the message communication in the whole constellation. Gateway has brought a bouquet of benefits for G2C and G2B services.

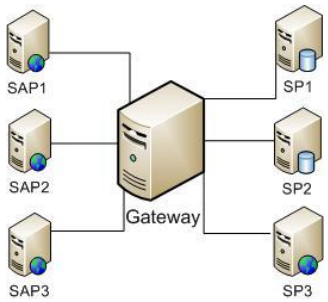

Figure 2. Gateway with SAP and SP

- **One Stop Service Access:** Integration of services across government departments in India give rise to provision of availability of services to citizens through single window.
- **Strong Authentication:** W3C Signature, and SHA-1 algorithm are chosen as the primary means of authentication for SAPs and SPs. W3C offers a strong binding of the SAP, SP to Gateway and helps in preventing security threats imposed by unauthorized users.
- **Nationwide Interoperability:** Gateway provides real time integration between systems and reuse of services where implementation requires no prior knowledge of the service.
- **Prevention of Data Loss:** Gateway being a message router, is not utilizing data communicated between SAP and SP. Data confidentiality, integrity is being ensured by means of encryption and digital signatures at XML block level.
- **Transaction Management:** Gateway keeps track of successful as well as unsuccessful messages which are routed through it, ensuring transaction logging and auditing. Each transmitted message is assigned with a unique audit-id to keep track of message delivery through the Gateway.

4. GATEWAY IMPLEMENTATION

Gateway implements eGov exchange standards for the purpose of message communication between SAPs and SPs. The standards define the common message structure with a provision of security, reliability etc. for government departments to communicate with each other through Gateway.

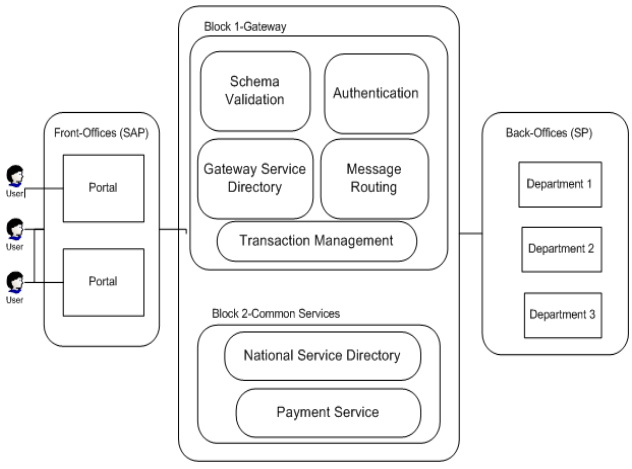

Figure 3. Gateway Structure

The overall functionality performed and the message structure for communication by this middleware is discussed in this section.

Figure 3 shows an overview of Gateway with major functional components and common services. Block 1 represents the components of the Gateway whereas Block 2 demonstrates the common services. NSD acts as a common service through which services attached with one Gateway can be accessed by other Gateway leading to constellation of Gateways. Common Payment service is a centralized component to make the payments for the services. Gateway implements various functionalities using major modules namely syntax and semantic validation, authentication and verification, message routing, transaction management and Gateway Service Directory (GSD).These modules are described briefly below.

- Syntax and Schema Validation- Messages received at Gateway will be checked for schema compliance as per the eGov exchange standards.
- Gateway Service Directory (GSD): It maintains information about the front end SAPs and back end (SPs) along with the services provided by them. If a service requested by SAP is not available in GSD of a particular Gateway then it can be further searched in National Service Directory (NSD) [15] for service availability with other Gateways.
- Message Routing –Message forwarding to the respective department in the back offices.
- Transaction Management – Messages flowing in and out of the Gateway are being audited with the help of unique audit-id. This id is being generated and assigned to the message as soon as any message enter or leaves the Gateway.

NSD is a centralized directory for all the services in the Gateway constellation. Modules shown in the Figure 3 are major functional components[1] of Gateway.

Message Structure

Messages structure defined for communication purpose is realization of SOAP structure but in a different manner. It introduces the capabilities of security, reliability etc. in the SOAP Body. W3C defines SOAP structure i.e. SOAP envelope which essentially encompasses SOAP body and SOAP header. 1) SOAP header as defined by W3C is an optional part in message communication and provides the capability of security, reliability etc. SOAP header is not being used in Gateway and capabilities provided by SOAP Header are built in eGov exchange standards [21] defined for communication purpose and will fall under SOAP Body. 2) SOAP body is the main message content that needs to be routed using SOAP protocol. It will be having message structure carrying authentication, routing information along with the contents provided by the departments i.e. SAPs and SPs. Gateway. Overall message structure defined is shown in figure 4.

Gateway will only utilize message content for routing and authentication purpose whereas data intended for SAP or SP department will only be read by respective SAP or SP, therefore Gateway is only acting as an intelligent router for inter-departmental communication. It is also possible to add multiple digital signatures for different XML blocks which can be intelligently routed to the respective departments or to the directories for locating the services. Digital signature at XML block level reduces the efforts required for signing the whole message. Introducing common message structure leads to syntactic and semantic interoperability.

[1] More detail about all components can be made available on request from researchers and developers.

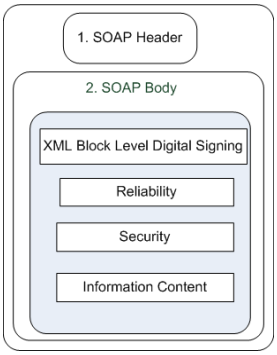

Figure 4. Message Structure

Technology Fall-out

Gateway is built using predominantly open source technologies and guarantees interoperability among government departments. Gateway application is written in Java. Linux is used as an operating system for hosting all components of Gateway. PostgreSQL is being used as database management system which is capable of handling large volume of data. JBoss application server is being used for deploying the Gateway. This is one of the largest implementation in India on open source technologies and provides performance of 250 messages/second with an average payload of 60 KB. Provided additional hardware infrastructure, Gateway is scalable up to 1000 messages/second.

5. CHALLENGES

We faced many technical and procedural challenges while designing, implementing and testing the Gateway. Some of them are discussed in this section.

5.1 Interoperability

Web services use WSDL to define basic structure such as data types, port bindings, message type, parameter binding etc. for establishing communication between client and server. However it does not enforce type system to be adopted for communication. XSD is used for defining type system and it can follow three design patterns [17]: (1) Russian Doll, (2) Venetian Blind and (3) Salami Slice. The recommended pattern because of strongly data type binding is Venetian Blind. Generation of web Service semantic through WSDL is easy but this approach ignores the design of the message schema. All the sender and receiver of inputs from web services should be capable of understanding the data type structuring as defined in the particular pattern. If the framework in which a particular web service is written is unable to understand the design pattern then there is a possibility of interoperability issues between different language frameworks. There are certain issues which we have observed while implementing & integrating Gateway with other applications.

- Null Value Unacceptable- There are certain data types that need to be handled cautiously while defining them in XSD. One of such data type is ‘dateTime’ where assigning null values are not acceptable for some of the language frameworks. Problem arises when the data type defined of type ‘dateTime’ expects the null values. A proper way of making the ‘dateTime’ element interoperable with other applications is to define such elements without null values.
- XML Array Incompatibility- Language framework in which a particular web service is being interpreted may not able to understand weakly-typed collection objects defined in XML Schemas and their mapping to the correct native data types.

XML interpretation of an array with null elements differs between various language frameworks. One to one mapping between native data types and XSD data types can lead to information loss during rendering.

- XSD: Any - XML schema exists as part of a web service, which sometimes leads to language binding. Sometime mapping of data type "xsd: any" becomes language dependent and it becomes necessary to understand the language binding [17]. It is always recommended to perform an appropriate mapping for XSD types i.e. to a value type in one language and to a reference type in another and it can be achieved by wrapping the value types in a complex type, marking complex type to be null for null values.

5.2 Testing

There are lots of issues faced while testing Gateway and few of them are discussed here.

- Manually generated test cases can determine quality issues but will not cover the testing in detail. In Gateway testing we have considered automation of negative and positive test case generation through tools that are web services oriented.
- Test Suite should have test cases for determining and testing the SOAP attachment size and its impact on performance.
- An abstract common data structure gets written in the form of XSDs and these XSDs are then hosted externally so that they can be re-used by WSDLs across the network. In most of instances WSDL definition import or include XSD schema within it through a URI. Common mistake performed in SOA testing is of testing the WSDL import within the local network, not on a wide area network, but sometimes the underlying schema URI becomes hard to find through the firewall. Gateway was tested extensively in internal as well as in external environment to avoid the WSDL schema import issues. Testing WSDL quality before development can also resolve the interoperability issues.

6. CONCLUSION AND FUTURE WORK

Gateway enriches distributed systems capabilities for providing transaction management, auditing, information management, computation, and system management. It also ensures scalability, interoperability, reliability and performance in the overall scenario. The use of mobile devices in providing e-Governance services is rapidly increasing, and the Gateway support use of such devices. Gateway has achieved the objective of single window delivery of services. It also demonstrates potential of open source technologies for developing such a comprehensive system that is capable of achieving interoperability and desired performance. The Gateway will undergo continuous improvement by incorporating additional features to it. Common services such as centralized payment service, which is in design phase, will be integrated into the system to facilitate faster and more convenient payment processing. Departmental workflow may be considered to be included in the Gateway. This middleware can be taken as hypothesis for integrating government applications on small as well as on bigger scale.

7. ACKNOWLEDGEMENT

We are thankful to all the members of Software Engineering group of C-DAC, Mumbai for their direct as well as indirect contribution for this paper. We would like to pay our special thanks to Mr. Zia Saquib, Dr. Padmaja Joshi and Dr. Shimon Modi for their valuable inputs towards this paper.

8. REFERENCES

[1] Bekkers V. 2005. "The Governance of Back Office Integration in E-Government: Some Dutch Experiences", *Electronic Government* ,2005, Berlin , Springer LNCS 3591,12–25

[2] Benamou N, Busson A, Keravel A. 2004. " Impact of e-Government Interoperability in Local Governments", *Electronic Government*, 2004, Springer LNCS 3183,82– 87

[3] Cava I, Guijarro L. 2003. " Interoperability Issues of Shared Infrastructures for E-Government", *Electronic Government, 2003*, Berlin, Springer LNCS 2739, 369–372

[4] Diedrich, E., Schmidt, D., and Wimmer, M. 2006. "A Three Dimensional Framework to Realize Interoperability in Public Administrations", *Workshop on Semantic Web for e-Government*, 2006, 54–59

[5] Fang, Z.2002. "E-Government in digital era: concept, practice, and development", *International Journal of the Computer, The Internet and Management*, 2002, vol. 10, no. 2, 1–22

[6] Heeks, R. (2001a), "Building e-Governance for development: A framework for national and donor action", *i-Government Working Paper Series Paper No.12*, University of Manchester, Institute for Development Policy and Management. Retrieved from http://idpm.man.ac.uk/wp/igov/igov_wp12abs.htm

[7] Janssen, M. and Scholl, H. J. J.2007. "Interoperability for electronic governance", *Proceedings of the 1st international conference on Theory and practice of electronic governance (ICEGOV)*, 2007,45-48

[8] Klischewski R. 2004. "Information Integration or Process Integration? How to Achieve Interoperability in Administration", *Electronic Government*, 2004, Berlin, Springer LNCS 3183,57–65

[9] Klischewski R, Scholl H.J.2006. "Information Quality as a Common Ground for Key Players in e-Government Integration and Interoperability" , *Proceedings of the 39th Hawaii International Conference on System Sciences,2006*

[10] Kook, Y.G., Lee, J., and Kim, J.S.2009. "E-Government Grid System Based on Multi-agent for Interoperability", *International Conference on Innovation management*,2009, 7-10

[11] Nikolopoulos, K., Patrikakis, C. Z., & Lin, B. 2004. "Forecasting systems for E-Government", *Electronic Government, an International Journal,2004*, vol. 1, no.4, 374 – 383

[12] Scholl, H. J. J.2005. "Interoperability in e-Government: More than Just Smart Middleware", *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS38), 2005*

[13] Scholl H.J., Klischewski R. 2007. "E-Government Integration and Interoperability: Framing the Research Agenda", *International Journal of Public Administration*, July 2007, Volume 30, Issue 8 & 9 , 889 – 920

[14] http://www.nsdg.gov.in

[15] http://india.gov.in/govt/national_egov_plan.php

[16] http://www.mit.gov.in

[17] http://www.ibm.com/developerworks/xml/library/ws tip-xsdcaution.html.