MIDDLEWARE AND TECHNOLOGY STANDARDS FOR E-GOVERNANCE

Jaijit Bhattacharya
India Research Lab, IBM,
IIT, Delhi, Hauz Khas,
New Delhi 110016
India
Email: bjaijit@in.ibm.com

KEYWORDS

Middleware, E-governance, standards, record management

ABSTRACT

Governments are moving towards ushering in an age of electronic governance or e-governance. However, they are facing the problem of investing in solutions that do not integrate with the rest of the systems or are not scalable and robust. This paper attempts to introduce the concept of middleware and technology standards as a tool to develop *integrable*, scalable and robust e-governance solutions, while employing multiple solution providers.

1. INTRODUCTION

Governments all over the world are trying to utilize IT for various purposes. The initial motivation usually comes from the need to improve efficiency of processes in the government. This may be concurred or followed by the second step comprising re-engineering of the processes. Another set of motivation may come from the need to provide various social services to citizens for improving the quality of life of the citizens. A third set of motivation may be to strengthen the democratic foundations of governance (opinion polls, voting etc.). These social services and democratic enablements correspond to the new activities that become economically viable due to the altered cost structure due to use of information technology.

Such complex requirements of electronic governance (Bansal and Bhattacharya 2000) pose two big challenges to the field of computer science. The first challenge is of managing the development of the solutions on a continual basis and then managing the large number of applications that need to interact with each other while maintaining security and privacy of the data. This needs to be accomplished in such a manner that change requirement in a single application should not trigger changes in other applications. Also, these applications may need to be developed in a massively parallel way unlike conventional development processes. Hence their development should be such that they are developed to be *integrable* i.e. from bottom-up, the design and development should be such that once they are ready, the applications automatically integrate with the rest of the solution and with future components. Also, the government should have the freedom to pick and choose the most appropriate application from any vendor and seamlessly plug in that application into the e-governance middleware and thereby, to the rest of the e-governance solution, making the government independent of a single solution provider.

The second challenge is of scalability, arising primarily from a need to maintain large number of records that may be created in geographically distributed data repositories. E-governance will spew petabytes of data, with trillions of records. The amount of information being handled by government is expected to grow up exponentially once e-governance is introduced. This is because the ease of transactions introduced by e-governance would encourage citizens and businesses to have more transactions with the government. In order to manage this enormous amount of data such that system performance does not get degraded, and such that the system is scalable, there is a need to automate the record management functions. Although there exists solutions for document archival system based on network-centric groupwares (Berchtold et al. 1999; Crespo and Garcia-Molina 1998) but they do not address the issue of archival based on policies of multiple applications (Batra et al. 2002). An example of such a policy is archiving be allowed only after auditing of the records has been completed. The solution should also be able to maintain the audit trail of all the records (Peha, J. M. 1999; Sandhu and Samarati 1996; Hansen 1983; Schneier and Kelsey 1999) at a required detailed level.

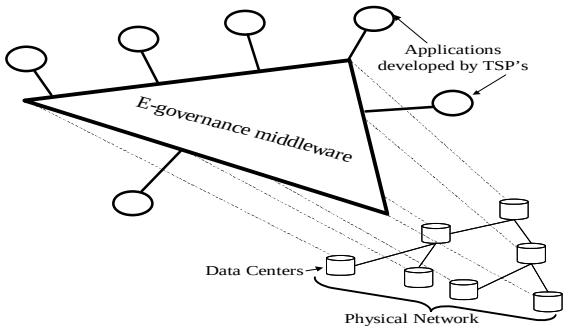
This paper describes some of the middleware components that would be required to meet the above challenges. It also discusses the standards required for developing a complex e-governance solution.

2. CURRENT SOLUTION TRENDS

Various governments, involved in the task of building an e-governance solution (Bansal and Bhattacharya 2000), are grappling with the problems of developing such a large system. One of the key problems is how to select and entrust a solution provider to deliver a particular component of e-governance. Given the numerous solution providers in the market with none having any experience in building a system as humongous as e-governance, this is a tough decision. However, a middleware that allows solutions of multiple vendors to be plugged in with ease, would solve this problem to an extent.

Another problem faced by governments is to contain costs by developing a portable/ replicable solution. The rationale behind such a solution is that, just as in businesses, around 85% of the processes are same across firms, within the same industry, it is expected that 85% of the processes should be similar across different governments. Thus, it should be possible to reuse the solutions developed for one government, for another government. Reusing the e-governance asset across different governments can substantially bring down the cost of developing e-governance solutions.

One option for tackling the above problems that is being considered by some governments, like the Government of Maharashtra, is to introduce a programming model consisting of a network, *Total Solution Providers (TSP)* and a middleware that can impose standardizations and extract the commonalties between different e-government applications (Bansal and Bhattacharya 2000). The network would consist of the physical connectivity to the administrative units, the data-centers and provide gateways for access through the Internet. *Total Solution Providers* or *TSP's* are solution providers who have domain expertise in some specific processes or departments of the government. Given their repeated exposure to the same processes, *TSP's* are expected to become efficient developers of applications for that particular domain and will maintain the applications, adapting them to changes in technology.



Figures 1: Conceptual Positioning of the Middleware

The middleware provides the glue between the network and the solutions developed by the TSPs (figure 1). Thus the middleware imposes the standards that allow the government to choose multiple solution providers for its various departments/ processes. This feature facilitates the solution to the problem of selecting multiple vendors. It also facilitates creation of an *integrable* solution.

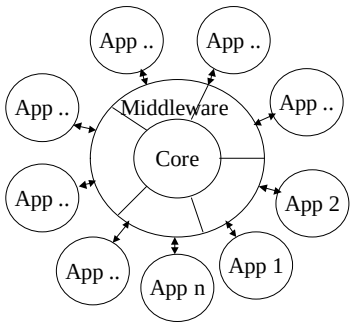
The rest of the paper assumes the above framework for delivering e-governance solution.

3. MIDDLEWARE TECHNOLOGY REQUIREMENTS

Technologies for a middleware solution for a comprehensive e-governance solution, that meets the objectives defined in the earlier sections, will have to address many diverse requirements that may be present due to various reasons. These reasons may be economic, political, technical and cultural amongst others. The requirements are classified into two categories, (a) middleware technology requirements that discusses the core technology requirements and (b) application requirements, which discusses abstraction of common code required for multiple applications/ departments.

3.1. Generic Middleware Requirements

The middleware should be able to support *phased implementation* i.e. it should be possible to have a unified approach but still implement the solution in phases. Since the government offices number in thousands (Bansal and Bhattacharya 2000) and are geographically distributed, it may neither be economical nor technically feasible to roll out the entire system together.



Figures 2: Middleware Structure

Hence the middleware itself needs to be deployable component-wise, having a core structure with peripheral components being added as and when required (figure 2). Thus all communication is routed through the middleware,

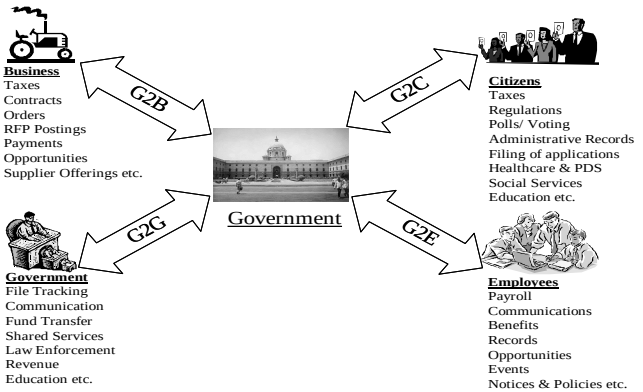
which brokers the communications and ensure access control (Tidswell and Jaeger 2000), security and privacy enforcement.

The middleware should support processes involving multi-department and multi-agency workflows. For this purpose, it is necessary that the different departmental offices (and also external agencies) are interconnected and share the same underlying back-end databases and applications. The middleware should also be able to facilitate integration with legacy systems.

Finally, the middleware should be capable of scaling with time in terms of number and complexity of applications, number of locations and number of users / usage. It should be reliable, that is, provide assured levels of service for uptime, availability and performance. The solution should also incorporate efficient back-up capabilities (Mohan and Narang 1993; Garcia-Molina and Ullman 2000) and the ability to handle contingencies and recover from failures.

3.2. Application Requirements

Unlike conventional middleware, which only support the basic technology requirements of applications that are to be developed using the middleware, a middleware for e-governance needs to also incorporate common code as part of the middleware itself. Such applications can be instant messaging, workflow etc. In addition to generic applications, the middleware should support interfaces to vertical-specific applications, which are government-to-business (G2B), government-to-employees (G2E), government-to-citizens (G2C) and government-to-government (G2G), as shown in figure 3. In addition, it should support citizen-to-citizen (C2C) applications also.



Figures 3: Interfaces for Key Government Verticals

The most important application requirement is that of security and audit trail (Schneier and Kelsey 1999). Various security services like authentication, multiple levels of access control (Tidswell and Jaeger 2000), confidentiality, privacy, data integrity (prevention of forgery) and non-repudiation need to be provided. In addition, ability to generate access logs, especially for sensitive data should be present (Tidswell and Jaeger 2000). The solution should integrate use of public key certificates and digital signatures, as applicable in India, in order to enable the transactions to have legal validity.

Next is the record management requirement as highlighted in the introduction section itself. The middleware should be capable of managing very large number of records as implied by the legal and accounting norms and practices. A prototype of such a system, called the *Policy Driven Data Administrator (PDDA)* has been developed and is seen as step towards building the complete middleware for e-governance (Batra et al. 2002). The discussion on *PDDA* system is beyond the scope of this paper.

Also, the middleware needs to provide a common human interface that needs to be simple enough to facilitate greater usage and that reduces the training costs. For citizen interfaces, this is all the more important where people may not be literate. Intuitive graphical user interface, use of speech and video technologies may be needed. Alongwith a simple human interface, the middleware should also be able to support local language interfaces for both employee and citizen applications.

Finally, the middleware needs to support a standard communication application and archiving (Arapis, C. 1999), which may include messaging, instant messaging, video conferencing and mission-critical inter-application communication.

4. STANDARDS

The key components of the technology standard that has been identified during interactions with various Indian state governments (Govt. of Maharashtra, Govt. of West Bengal, Govt. of Arunachal Pradesh, Govt. of Uttaranchal) are (a) databases, (b) operating systems, (c) schema and nomenclature standards, (d) middleware standards and (e) security standards. The technology standards help in building a solution within a framework. This framework is the architecture framework for the complete integrated solution across the state departments.

4.1. Architecture Standards

The solutions options for architecture standards are 2-tier, 3-tier or n-tier. In late nineties three or n tier architecture came to be used along with object orientation. The advent of platform independent languages like Java and browser based Internet technology further strengthened this approach. These architectures separated

the applications into presentation, business and data tiers for better flexibility, maintainability, performance and scalability. The presentation layer without business logic becomes thin and uses browser/Java technology to become platform independent. The business and data layers are generally hosted on separate servers. The business logic is typically hosted on system software called *Application Servers* providing common services e.g. security, locking etc. Because of the advantages of 3-tier architecture in a complex development environment, this architecture appears to be the preferred architecture standard. Hence any e-governance solution needs to be based on a 3-tier architecture (where individual tiers need not be physically separate tiers) with deployment of distributed components, which can communicate across tiers.

One of the important requirements is that the architecture must be open and should allow interoperation with various standard products in the market. Hence, the solution needs to be based on open standards supporting HTML, XML, WML, HTTP, TCP/IP, SSL, SET, PKI, X.509v3, LDAP, Java, Servlets, JSP, EJB, Enterprise Java APIs (JDBC, JMS, JTS, JNDI etc.), CORBA, IIOP, IMAP4, POP3, CWMI, SOAP, UDDI etc.

4.2. Technology Standards

Based on interactions with various Indian state governments, certain key parameters have evolved for determining the technology standards to be adopted. These parameters are (a) applicability of the technology, (b) scalability, (c) robustness, (d) availability of relevant skilled manpower, (e) vendor commitment and availability of vendor support and (f) cost of ownership. Before adopting any standard, the standards need to be evaluated against the above parameters. The evaluation of standards is beyond the scope of this paper.

Database Standards

Given the complexity of the solution requirement, any database chosen for the solution, needs to support full parallelism without any restrictions on Update /Insert /Delete, specially on LOBs. The database would also need to directly support Recursive SQL (and not through programming), in order to maintain the efficiency of the database. The database should also be devoid of any resource bottlenecks for efficiency reasons. The database also needs to have a robust *Cost Based Optimizer* for the same reason (Garcia-Molina and Ullman 2000). Because of the distributed nature of the solution, the database needs to be based on the *Shared Nothing Concept* such that one lock manager serves one database node only.

Since it is expected that e-governance solutions will have long running transactions, hence the database needs to have a log file architecture without the need for rollback segments so that the rollback segments do not get full forcing the work to be rolled back (Garcia-Molina and Ullman 2000). Also, since one cannot stop the long running transactions, it is imperative that the database supports efficient online backup and restore. It would also need to support optimal data buffering through unlimited number of bufferpools. The database also needs to support real data growth onto parallel servers for workload distribution.

Along with databases, there is a need to standardize the OLAP software also. The chosen OLAP needs to have multidimensional analysis capabilities, with support for a large number of dimensions. Moreover, given the varied sources of information, the OLAP should be able to access data from relational data source, spreadsheet and text files. It should not require an RDBMS at the backend to build, run and operate the multidimensional database. Given the sensitivity of data, the OLAP needs to provide high level of security till the cell level. Given the diversity of platforms, the OLAP needs to be available on multiple platforms - Unix, NT, AS/400, S/390, clients on Windows, Mac and Unix. Finally, for the tool to be used by a large section of people, the OLAP would need to allow multidimensional analysis to be web enabled.

Operating Systems

The choice of the operating system (OS) is critical for the success of any e-governance solution. The OS needs to be stable, secure, scalable, open and cost-effective. The essential features of OS are security (secure from hacking and viruses), vendor independence (so that no one vendor can hold the government to ransom), application portability, skills availability, future survivability of the OS and support to the OS from the IT community.

Schema standards and nomenclature standards

The need for such a standard arises because of the involvement of multiple developers. Standardizing schemas and nomenclatures brings down the cost of development and subsequent updgradation and maintenance. Detailed study of few of e-governance solution requirements needs to be done before this set of standards can be prescribed. Since multiple solution providers are typically involved in an e-governance solution, hence these standards can be arrived at only after discussions with the application developers.

Middleware Standards

Middleware needs to provide services such as identification, authentication, authorization, directories, and security to all applications. By promoting standardization and interoperability, middleware will make advanced network applications much easier to use. The key middleware components are (a) Web Application Server, (b) Inter-application communication and

messaging, (c) Mailing and Collaboration software, (d) Language and data interchange standards.

Security Standards

Security is critical for the running of any e-governance solution. Security is enforced through multiple components such as firewall, authentication & authorization mechanism, and audit control mechanisms (Schneier and Kelsey 1999). It needs to provide a secure, automated and role-based, policy-based user management (Bonatti et al. 2000; Sandhu and Samarati 1996; Batra et al. 2002). It should be able to centrally define and manage security policy for a broad range of e-governance and other applications. It would also need to have role-based administration model for delegation of administrative privileges and group users according to business needs. Security would also need to have a workflow to accommodate multilevel approval hierarchies and it should be configurable to the local government/departmental environment, planning system, or other workflow products to collect and process information from the various touch points throughout the government. Security also includes PKI enablement for existing Web-based application. It would need to support authentication and access control for web-browser user through Used IDs and passwords, client-side certificates, or RSA secured ID tokens (Tidswell and Jaeger 2000).

5. CONCLUSION

E-governance solutions are complex and expensive solutions that would need to be built brick by brick over a period of time, involving multiple solution providers. In order to have a successful e-governance solution, governments need to adopt middleware standards that are common across the entire e-governance solution. In addition to middleware standards, governments need to adopt certain technology standards. Such middleware and technology standards enable development of *integrable*, scalable and robust solutions and cut down the cost of development and maintenance of e-governance solutions.

REFERENCES

Arapis, C. 1999. "Archiving Telemeetings". In Proceedings of the eighth international ACM conference on Information knowledge management.

Bansal, V. and J. Bhattacharya. E-governance solution for government of Maharashtra. Technology whitepaper, India Research Lab, IBM, 2000.

Batra, V.; J. Bhattacharya; H. Chauhan; A. Gupta; M. Mohania; U. Sharma. 2002. "Policy Driven Data Administration". In POLICY 2002, IEEE 3rd International Workshop on Policies for Distributed Systems and Networks

Berchtold, S.; A. Biliris; E. Panagos. 1999. "SaveMe: a system for archiving electronic documents using messaging groupware". In Proceedings of the ACM international joint conference on Work activities coordination and collaboration.

Bonatti, P.; S. di Vimercati; P. Samarati. 2000. "A modular approach to composing access control policies. In Proceedings of the ACM Conference on Computer and Communications Security"

Crespo, A. and H. Garcia-Molina. 1998. "Archival Storage for Digital Libraries". In Proceedings of the third ACM Conference on Digital libraries.

Garcia-Molina, H. and J. D. Ullman. 2000. "Database System Implementation", Prentice Hall.

Hansen, J. V.1983. "Audit considerations in distributed processing systems". Communications of the ACM, (ISSN: 0001-0782), Vol 26.

Harrison, M. A.; W. L. Ruzzo; J. D. Ullman. 1976. "Protection in operating systems". Communications of the ACM, 19(5):236-242.

Mohan, C. and I. Narang. 1993. "An efficient and flexible method for archiving a data base". In Proceedings of the ACM SIGMOD international conference on Management of data.

Peha, J. M. 1999. "Electronic Commerce with verifiable audit trails". In 9th Annual Conference of the Internet Society, INET.

Sandhu, R. and P. Samarati. 1996. "Authentication, Access Control, and Audit". ACM Computing Surveys, 1996

Schneier, B. and J. Kelsey. 1999. "Secure audit logs to support computer forensics". In ACM Transactions on Information System Security.

Tidswell, J. E. and T. Jaeger. 2000. "An access control model for simplifying constraint expression". In *Proceedings of the 7th ACM conference on Computer and communications security, Pages 154 – 163.*

AUTHOR BIOGRAPHY

JAIJIT BHATTACHARYA was born in India and obtained his B.Tech in Electrical Engineering in 1995 from the Indian Institute of Technology. He did his MBA in Systems and Finance from the Indian Institute of Management, Calcutta. He worked for Accenture (formerly Andersen Consulting) for more than five years in the period of which he was involved in implementation of very large systems in over nine countries. He has also developed business models and strategies for leading companies in the IT, media and computer hardware industries. He is currently doing research in e-governance systems and web mining in the India Research Laboratory, IBM. He is also pursuing his PhD in Computer Science from the India Institute of Technology, Delhi. Jaijit speaks five languages including English, French, Bangla, Hindi and Bahasa Indonesia.