TOWARDS AN E-GOVERNMENT LAB

Marijn Janssen and Corné Versteegt
Faculty of Technology, Policy and Management,
Delft University of Technology,
Jaffalaan 5, 2628 BX, Delft, The Netherlands,
Tel. +31 (15) 278 1140, Fax. +31 15 278 3741,
E-mail: marijnj@tbm.tudelft.nl, cornev@tbm.tudelft.nl

KEYWORDS
Simulation, e-government, animation, components, decision support

ABSTRACT

E-government is emerging as the new way of coordinating relationships with business and citizens as the result of technological developments As such it requires the cooperation between many government organizations, the redesign of business processes, functions and responsibilities and the identification of opportunities enabled by information and communication technology. New government concepts need to be developed, tested and imbedded in the existing ICT-architectures The goal of this paper is to identify the foundations for an 'e-government laboratory' and to explain the crucial role of simulation. In this paper the ideas behind the e-government laboratory are presented, the role of simulation is explained and a pilot project is presented. The pilot project is a simulation study to visualize the opportunities and implications of an open, flexible approach for service provision.

INTRODUCTION

E-government is emerging as the new way of coordinating relationships between government organizations and its customers, which can be business and citizens. The driving force behind e-government is the rapid application of Internet technologies to increase the interoperability between government, businesses and citizens. Governmental organizations are challenged to provide more customer-oriented products and services and cross the distance with customers. Customers can be targeted through multiple channels, such as web-based, call centers and physical offices in the municipal hall. Businesses and citizens should be able to participate in policy development using all kinds of groupware.
E-government has implications in four areas, (1) policy development, (2) policy execution, (3) service provision and (4) policy enforcement (Janssen and Wagenaar 2002). Although e-government has great promises, the application of new ideas and concepts in those four areas stays far behind. In order to exploit these opportunities in an efficient and effective way, the need to restructure administrative functions and processes is clearly felt. The government faces a major challenge as 85% of all public sector information and communication technology (ICT) project are deemed to be failures (Ramsey 2000). 25-50% of the project are never implemented or abandoned immediately after implementation. The main problem seems to be the difference between design ideas and organizational reality (Heeks 1999). An *e-government laboratory* (e-gov lab) should be able to overcome this gap and provide insight into new organizational forms and how government can implement ICT in its organization. Simulation plays a crucial role in this lab.
The *goal* of this paper is to identify the foundations for an e-gov lab and to explain the crucial role of simulation within such a e-gov lab. The faculty of Technology, Policy and Management at Delft, University of Technology initiated a e-gov lab for identifying, testing and evaluating new governmental concepts. This e-gov lab is aimed at bringing governmental representatives, its customers and ICT companies together, providing insight into and evaluating various kinds of e-government concepts and to identify the steps necessary for fitting the concepts within government organizations. In this way it should increase the speed of the adoption of e-government technology.
This paper starts with describing e-government and the problems around the adoption of innovative government concepts. Thereafter a design is given of the e-gov lab. A pilot study at the Dutch municipalities is described which was aimed at evaluating the concepts behind the e-gov lab and to extend them. At the end of the paper we draw some concluding remarks considering the future development of the e-gov lab.

E-GOVERNMENT

Electronic government is a field springing from pressing economic needs and political initiatives (Grönlund 2002). E-government can be defined in various ways. An overview of definitions of e-government can be found in Janssen and Wagenaar (2002). The main characteristics mentioned in these definitions are the following.

- Internet-based technologies as enabler;
- Integration of strategy, process, organization and technology;
- Improving the efficiency, transparency and customer orientation of government operations;
- Deals with relationships between government organizations and its customers;
- E-government demands a transformation of current operations. Internal organizations as well as the external environment are involved in this transformation.

In Figure 1 the main actors and their relationships are shown based on Grönlund (2002). Arrows indicate influence, and circles indicate domains of control. Generally speaking, citizens elect politicians, politicians control the government and government provides services to citizens and business. There are a number of relations, each node in the system influences both the others by a number of relationships, and all nodes are interrelated in a complex pattern.

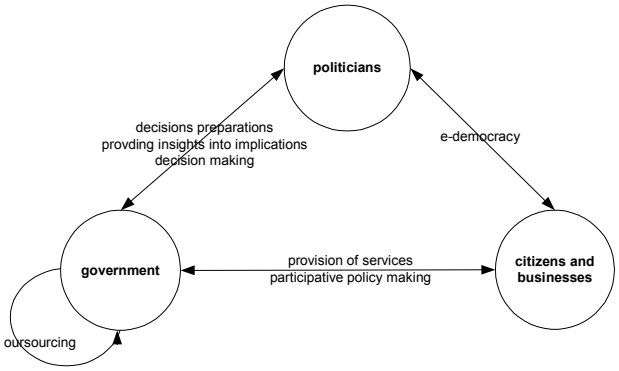

Figure 1: E-government Relationship

Figure 1 shows the possible areas for transformation of relationship between actors. Much of e-government is about the delivery of services and the government's dealings with the private sector. But there is another dimension to it, as boundaries between departments, business and citizens are often vague. Citizens are also an owner or shareholder of government itself. In the digital age, people have an ability to communicate, to participate and add value. Business and citizens demand a more active role in the policy development process and want to cooperate with politics in order to influence decision-making. During policy execution citizens and business want to be involved, and want to be part of the process. During and after execution government has to provide information about the status and progress for public accountability and transparency of government functioning.

The business engineering process of e-government is a complicated endeavor. Crucial for success during the development of innovative e-government concepts seems to be the bridging of the communication gap between all kinds of actors including politicians, systems developers, citizens and businesses. This step includes the translation of e-government concepts into business processes and applications by defining one or more feasible projects. Many of the solutions are technical in nature, but ultimately require the change of organizational structure of government. The modeling and especially visualization of concepts is a crucial factor to provide insight into the concept for high-positioned government officials.

FOUNDATIONS OF AN E-GOV LAB

In this section we present our initial thoughts regarding the structure of the e-gov lab. The e-gov lab should be a neutral place where stakeholders from science, government, technology and service providers can come together, discusses innovative e-government concepts and test them on organizational and technical feasibility. Animation and gaming can be used for providing insight into e-government concepts, performance indicators can be used to evaluate the added value of hypothetical concepts and simulation output and input can be used for controlling and testing e-government architectures.

The e-gov lab is focused on the transition towards more open and flexible ICT-architectures. Applications will be more fine-grained and similar functionality can be provided by applications or services from various parties. Futures applications will be structured in small simple units, which co-operate through rich communication structures and information gathering.

Component-based applications are rooted in the object-oriented modeling approach (Fan et al. 2000). Objects are straightforward abstractions of real-world entities and have properties such as encapsulation, inheritance, and polymorphism. Objects greatly increase software reusability and simplify the software development process (Fan et al. 2000). The component model, on the other hand, focuses on building information systems by combining and matching pre-developed software objects, i.e. components. The focus is not on the properties of objects but on the combination and integration of different software components. The manageability increases as large components can thus be constructed from smaller components. In essence, a complex problem is split up into smaller problems, which can be solved independently. Each single component can be replaced by another component without affecting the others. Components can run from different computer platforms and interact with each other through standardized interfaces.

Service-based application means that services are configured to meet a specific set of requirements at a point in time, executed and then disengaged. Services only 'exist' during execution. This is closely related to the definition of a service in service marketing: "an act or performance offered by one party to another. Although the process may be tied to a physical product, the performance is essentially intangible and does not normally result in ownership of any of the factors of production" (Lovelock et al. 1996).

Applications can go a step farther and incorporate characteristics such as intelligence, goal driven behavior and learning abilities. Characteristics commonly viewed as that make up a *software agent* (Etzioni and Weld 1995, Nwana 1996). Software agents help people with time-consuming activities by gathering, analyzing of information, making decisions and act on those decision.

The e-gov lab will be designed around a middleware architecture, which receives input and output from the environment. The environment consists of simulation and software components. Both can communicate with middleware using messaging technology or using web-services.

The requirements on the e-gov lab can be summarized as follows.

1 Various architecture levels such as organization, business process, application and network should be visualized;

2 Interactions and dynamic behaviour between and within various levels of architecture should be visualized. Especially the interactions between components and the relationship with the business processes should be visualized;

3 Various stakeholders perspectives should be modelled in order to enable communication between them, including perspective for high-positioned e-government representatives and technology experts;

4 Modelling should capture the requirements discussed above, but avoid presenting so much detail that it requires too much effort and scarce time of decision-makers to understand the models.

5 Provide insight into the current 'as-is' as well as potential future 'to-be' situations;

6 Should support the testing of components and the fit within the existing application architecture;

7 Support stakeholders to draw conclusions about the benefits and disadvantages based on the insight gained.

Essential is the modeling of the various architecture layers and the interaction with real software. The *key issue* in modeling is the choice of phenomena to include and to omit (Janssen 2001). Modeling should capture the requirements discussed above, but avoid presenting so much detail that it requires too much effort and scarce time of decision-makers to understand the models.

ICT-Architecture is the description of the set of components and the relationships between them (Armour et al. 1999). Architecture is aimed at coordinating various architecture layers, including hardware, network, system, application and enterprise level. The relationships between architectural levels should also be taken considered for a complete architectural picture. Generally speaking, this research is about modeling architecture layers and the dependencies between these layers.

The bridge between policy and systems developers should be bridged during development of innovative e-government concepts. Many of the solutions are technical in nature, but ultimately require the change of government. The *visualization* of concepts is of high importance in order to provide insight into the concept for high-positioned government officials.

APPLICATION ARCHITECTURE

The foundations of the e-gov lab need to be translated to an environment supporting these requirements. The application architecture of the e-gov lab is schematically shown in Figure 2. The core of the application architecture consists of the simulator and middleware, communicating with each other. The simulator visualizes the ICT-architecture and performance of the ICT architecture. A user-interface enables input from actors to the simulator.

The middleware can send and receive in and outputs from software components, services provided by third parties and the simulator. The components can communicate with the architecture using messages or web-services over SOAP. The middleware translates all incoming message into XML format, outgoing message can be translated from XML into the format desired by the application. In this way communication with components not supporting XML is possible.

The middleware architecture can be used to control and test components or services. Standard or dedicated components or services are procured off the shell to select the best-of-breed software. The components are integrated with the rest of the systems, preferably automatically. Software suppliers can either assemble their component out of existing ones, or develop and evolve atomic components. In the future the middleware might provide a 'habitat' for autonomous agents in order to evaluate the possibilities for software agents in e-government.

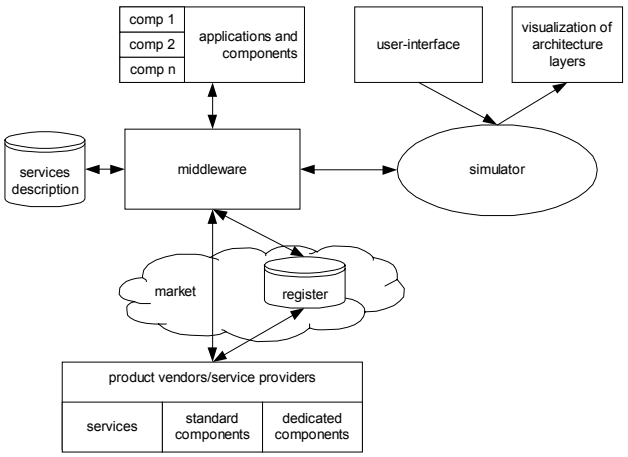

Figure 2: Application Architecture of the E-Gov Lab

Services descriptions are necessary for defining services and comparison with the descriptions offered in the markets. Web Services Description Language, WSDL, can be used to describe the properties of services. A register is necessary to discover and integrate components and services. These could be based on UDDI (Universal Description, Discovery and Integration), or provided by intermediaries using other standards. Public registers often fail to provide the necessary trust and reliability, therefore, it is not unlikely that intermediaries will com into existence providing services, including an information aggregation trusted, matching and technical facilitating roles (Janssen 2001).

In the future software will be shaped by open, flexible, robust and living architectures consisting of active components. Agents characterized by autonomous, evolving and negotiating attributes matching living organisms attributes propose a way to work in an uncertain and continuously changing environment (Bradshaw 1997). In the future the e-gov lab should support this kind of architectures.

Simulation can be used to understand the behavior of a concrete system, to evaluate various strategies for the operation of the system, and to study the impact of scenarios representing a particular path to a hypothetic future situation (Shannon 1975). The simulator is the core of the application architecture and used to simulate and visualize the various architecture layers, to interact with the middleware and enables humans to interact. The roles of simulation in the application architecture are:

- Evaluation of courses of actions using 'what-if' analyses;
- Visualization of various architecture levels;
- To provide insight into e-government situation for government representatives and other stakeholders;
- Control of components and applications, by sending and receiving messages;
- Testing applications; performance and failure tests;

PILOT PROJECT

Now we have discussed the foundations of the e-gov lab we will evaluate and try to extend them using a pilot project. This pilot project is aimed at verifying the concepts proposed in the preceding section and to sharpen the requirements. The pilot project is about the identifying of components and testing the fit of these components within the existing application architecture of municipalities.

Dutch municipalities are free to design their information architecture and to choose appropriate software vendors. Within municipalities there is no central management and departments can buy their own applications for each process. As a result, municipalities have a highly fragmented ICT-architecture, consisting of legacy systems for each product they offer. These systems are often monolithic packages and thus extremely difficult to reconfigure and to integrate with applications developed by other vendors and to provide access to new distribution channels. The existence of isolated, overlapping in function and content, highly fragmented and unrelated computerized applications within the same public organization has resulted in a major *interoperability problem* and has led to 'isolated islands of technology' while information systems were viewed as being internal to the public organizations (Fan et al. 2000). As technology continues to evolve at an accelerating rate, nontrivial hard- and software will remain diverse and heterogeneous.

The cross-municipality information managers' council initiated the "*AnalysePilot*" aimed at developing a reference architecture that should provide guidance for the development towards a component-based architecture. Such an architecture should not only bring online the 290 products currently provided in the municipalities' portfolio, but also support existing distribution channels. One of the goals of this project was to support management in their decision-making about the potential of a component-based architecture. When this decision-making turned out to be positive, a follow-up project should to be started for developing a prototype of such architecture to prove the concept in practice.

The simulation is aimed at showing the business case for an open, flexible component-based ICT architecture supporting multi-channel service provisioning. Simulation is used to model the existing information architecture by visualizing the organizational process, the interactions with legacy system and the technical infrastructure. For this purpose the existing process and application

architectures were analyzed within three municipalities, Almelo, Breda, and Zaandstad. Based on these detailed architecture a generic architecture covering the details was derived and a simulation of this architecture was build. In the following step a number of possible future business processes were derived. Based on the existing and future processes and existing application architecture a number of components for providing functionality for theses processes was derived. This was a highly intuitive process largely dependent on the knowledge of the experts involved.

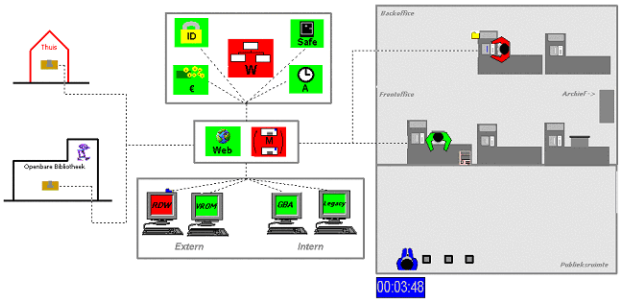

Figure 3: Screenshot of the Animation

A simulation including a component-based architecture was developed. The functions of the components were simulated to ascertain that the model was independent of specific implementations. Municipalities should be able to replace any component with better ones when necessary, consequently only the functions of the components were simulated. This simulations shows that components and legacy applications developed at different time periods and by different software vendors can work together. So, investments in legacy systems can be protected. By just plugging in new components changes can be made more easily.

Two new channels were added to the simulation, citizens using a computer at home and connected to the Internet requesting a driver's license and citizens using a terminal in the library to request a driver's license. The simulation shows the interactions over time to and from these channels using a component-based architecture including wrapped legacy systems. It shows that on a conceptual level a component-based approach is flexible and open enough to support a multi-channel approach. Probably even mobile channels can be supported using this approach.

The pilot study was used to evaluate the foundations of the e-gov lab and expand them. We could draw a number of conclusions with regard to the concepts. Simulation plays a crucial role in the e-gov lab. There is a need for modeling various architecture layers and that there is a need for a close connection to the existing architecture of government. A requirement is that legacy system should be tested, as it might be impossible to let them work with other information systems. Another requirements is that some software components might need to be simulated while others should be part of the e-gov lab and receive inputs and provide outputs for the other components.

CONCLUSIONS AND FURTHER RESEARCH

In this paper the main foundations of an e-gov lab were discussed. We gave an overview of the main problems with developing and implementing e-government concepts. Thereafter we discussed the main foundation of the e-gov lab. A pilot study was performed to test the concepts and can be seen as a first pilot study

The e-gov lab provides a neutral place for stakeholders to identify and evaluate new e-government concepts and to test the application architecture. Simulation plays a crucial role as it can be used to visualize architecture layers and the dependencies between these layers. Communication to various stakeholders with various levels of ICT knowledge seems to be crucial. Modeling should capture the various architecture layers and have a close connection to the already existing architecture. The simulation should be connected to an application architecture in order to support the evaluation of more flexible, open and intelligent application architectures. A requirement for testing the architecture is that new services, components, but also legacy system should be tested, as it might be impossible to let them work with other information systems.

REFERENCES

Armour, F.J. Kaisler, S.H. and Liu, S.Y. 1999. "A big-picture look at Enterprise Architectures." *IEEE IT Professional* 1, No. 1, 35-42.

Bradshaw, J.M. 1997. *Software agents*. MIT Press, Menlo Park.

Etzioni, O. and Weld, D.S. 1995. "Intelligent Agents of the Internet: Fact, fiction, and forecast". *IEEE Expert* 10, No. 4, 44-49.

Fan, M., Stallaert, J. and Whinston, A.B. 2000. "The adoption and design methodologies of component-based enterprise systems." *European Journal of Information Systems* 9, No. 1, 25-35.

Grönlund, Åke. 2002. *Electronic Government: Design, Applications, and Management*. Idea Group Publishing,.

Heeks, R. (1999). *Reinventing Government in the Information Age*, Routledge, London.

Janssen, M.F.W.H.A. 2001. *Designing Electronic Intermediaries. An agent-based approach for designing interorganizational coordination mechanisms*. Delft, The Netherlands: Doctoral Dissertation Delft University of Technology.

Janssen, M. and Wagenaar R.2002 *Trendwatching e-government: Concepts, state of the art and developments*. Report Ministry of Transport, Public Works and Water Management (in Dutch).

Lovelock, C.; S. Vandermerwe; and B. Lewis. 1996. *Services Marketing*. Prentice Hall, Europe.

Nwana, H.S. 1996. "Software Agents: An Overview." *Knowledge Engineering Review* 11, No. 3, 205-244.

Ramsey, T. 2000. What a difference in 10 years! *IBM Government Journal*.

Shannon, R.E. 1975. *Systems Simulation: The art and science.* Prentice Hall, Englewood Cliffs, New Jersey.

AUTHOR BIOGRAPHIES

MARIJN JANSSEN finished his doctoral dissertation in 2001 on electronic intermediaries that support the coordination of activities between buying and selling organizations. Currently, he is a member of the Information and Communication Technology Group of the Faculty of Technology, Policy and Management of Delft University of Technology. His email and web-address are marijnj@tbm.tudelft.nl and <www.tbm.tudelft.nl/webstaf/marijnj>.

CORNÉ VERSTEEGT is a researcher in the Systems Engineering Group of the Faculty of Technology, Policy and Management of Delft University of Technology. He specializes in logistics and control systems. Currently, he is finishing this dissertation. The topic of his dissertation is holonic control of large-scale automated logistic systems. His email and web-address are cornev@tbm.tudelft. and <www.tbm.tudelft.nl/webstaf/cornev>.