

E-government reference model

Concept paper

Alexander SAMARIN
SAMARIN.BIZ
alexandre.samarine@gmail.com

Abstract — An e-government reference model is proposed to optimise the implementation of e-government within a country as well as among several countries.

Keywords — reference model, e-government, architecture

I. INTRODUCTION

This concept paper describes an e-government reference model which is presented from the several points of view.

II. PARTNER – GOVERNMENT INTERACTION VIEW

Four possible types of interactions between the government and citizens, local businesses and other organisations (briefly, partners) are shown in figure 1.



Fig. 1. Four types of interactions

These types of interactions are:

1. The government sends an announcement, e.g. that a law has been changed.
2. A partner sends a declaration (also some kind of announcement) to the government, e.g. that this citizen has changed his/her address.
3. The government demands a partner to do something, e.g. to pay taxes.
4. A partner demands the government to do something, e.g. to provide a fishing certificate.

The last two types of interactions can be long-running interactions – there may be some noticeable time (weeks) between sending a demand (start) and receiving the result (finish). Even, the partner and the government may interact between the start and finish as shown in figure 2.

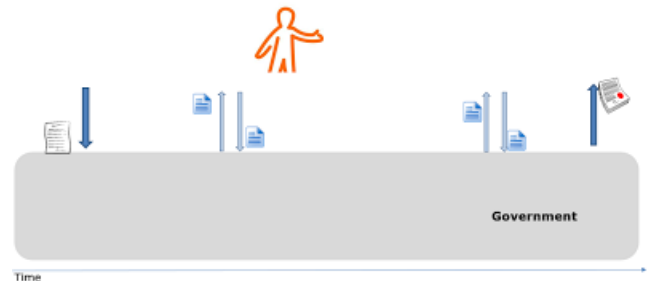


Fig. 2. Long-running interaction

In addition, the partner may have to deal with various entities within the government (e.g. different ministries). Usually, each ministry works in own way thus complicating life of the partner as show in figure 3.

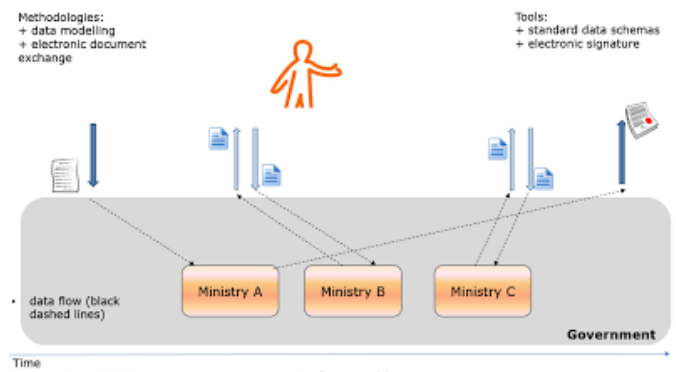


Fig. 3. Long-running interaction may involve several governmental entities

To protect the partner from the internals of the government and to unify his/her interactions with the government, the e-government acts as a shell which coordinates the flow of data (and documents) between the partner and the government. The e-government treats all governmental work as processes. These processes fulfil their goals by coordinating various services. For example, a partner's demand can be a result of joint work of three ministries in four top-level steps as shown in figure 4. The blue circled numbers show the process (flow of control) steps.

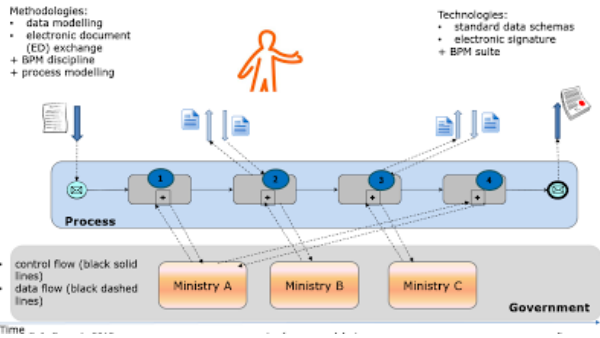


Fig. 4. Example of a top-level process

The related flow of data (and documents) is shown by the red circled labels in figure below.

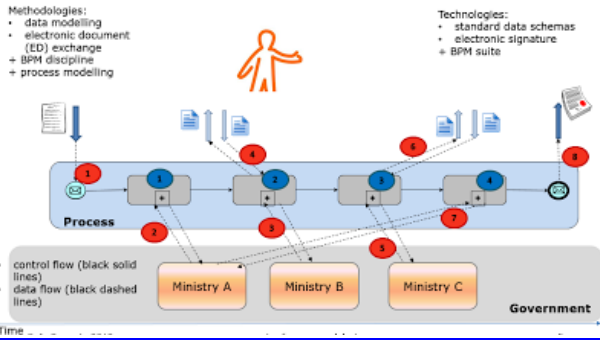


Fig. 5. A top-level process

Each top-level step in figures 4 and 5 is actually a process fragment which may be rather complex. For example, the details step #2 are showed (see figure 6) that one of its activities is a task for the partner (activity 2b).

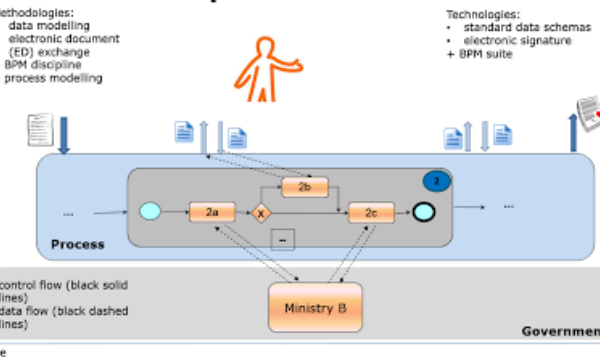


Fig. 6. Details of a step in the top-level process

A possible sequence of the execution of this fragment is shown in figure 7 by the red circled numbers.

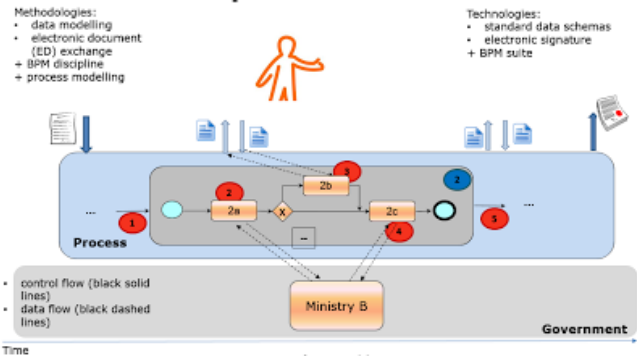


Fig. 7. Details (with dynamics) of a step in the top-level process

The usage of processes simplifies:

- notifying the partner about the progress in processes related to him or her;
- monitoring of service level agreements, and
- continual improvement.

To streamline the work of partners with the government, e-government provides a social collaborative extranet for all partners. This extranet:

- helps partner to manage all electronic documents (which are exchanged between the partner and the government) in a secure manner;
- helps partner to execute his/her tasks, and
- allows a partner to interact with other partners.

This extranet is an interface layer between the partners and e-government (thus with the whole government as well) as shown in figure 8.

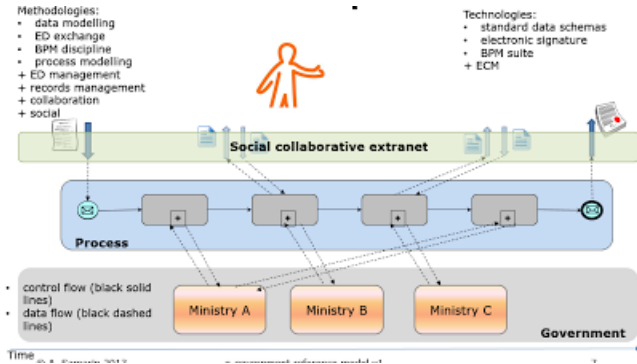


Fig. 8. Position of the social collaborative extranet

III. PARTNER VIEW

For the partner, the social collaborative extranet may have the following visual design (see figure 9).

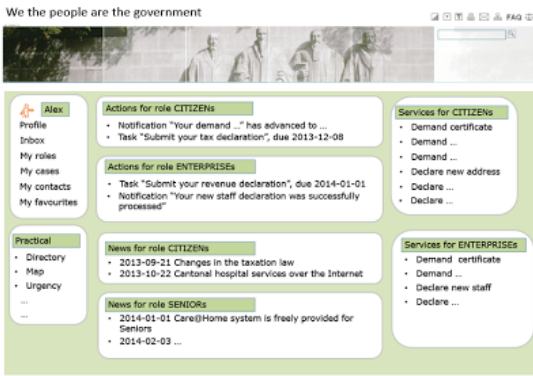


Fig. 9. Example of the partner's view

This extranet considers that:

- Each partner has several roles, e.g. YOURSELF (person and his/her legal representatives), CITIZEN (person officially leaving in this country), ENTERPRISE (manager of a business), SENIOR (persons with age over 60 years), etc.
- Person may select which roles he/she is carrying out at a particular moment in time.
- Each communication between a partner and the government is a case with associated documents, data, audit trails, records, service level agreements and key performance indicators.
- A case may be completed or on-going.

IV. THE GOVERNMENTAL ENTITIES INTEGRATION VIEW

E-government, which acts also an inter-ministerial coordination tool, ultimately resolved the integration problems within the government. Instead of that ministries are connected to each other in ad-hoc way, the e-government offers an integration process which delivers data and documents between all ministries in a systematic way as shown in figure 10. It is actually a centralised service (backbone) for the inter-ministries secure electronic exchange (like sending / receiving registered letters).

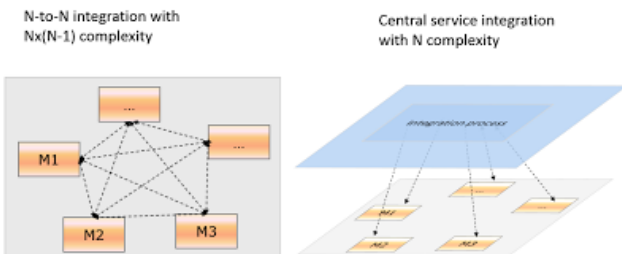


Fig. 10. Reducing complexity in integration between governmental entities

Generally, the backbone is decoupled from intra-ministry applications through two adapters: dispatcher (handle messages coming from the backbone) and expeditor (handle messages going to the backbone). To be transmitted through the backbone, each message (business data and documents) is

protected by three "envelopes" (marked by blue circled number in figure 11):

1. Business (processing) envelope
2. Delivery (addressing) envelope
3. Transportation (routing) envelope

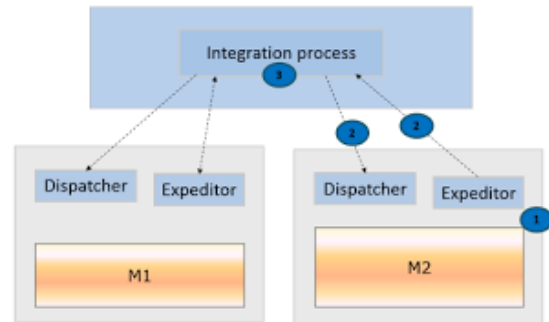


Fig. 11. Integration process between governmental entities

Of course, the access to open reference data (e.g. list of addresses in the country, some geodata, etc.) is different.

V. APPLICATION ARCHITECTURE VIEW

In general, the e-government application architecture is as in figure 12. There are three main technologies:

- Enterprise Content Management (ECM) for the social collaborative extranet;
- Business Process Management (BPM) for coordination and integration backbone, and
- Service Oriented Architecture (SOA) for coordination and integration backbone.

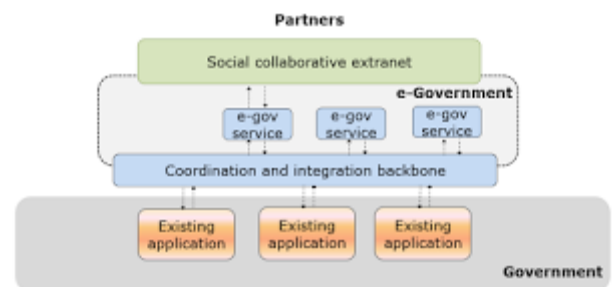


Fig. 12. Application architecture overview

Let us put this application architecture in the context. For the long time, e-government application architecture was portal-centric and its applications (blue "emabas") were extensions for some internal applications as show in figure 13.



Fig. 13. Application architecture – portal-centric stage

The proposed application architecture is actually, the introductory architecture which introduces necessary flexibility. E-government applications may span several existing applications as shown in figure 14.

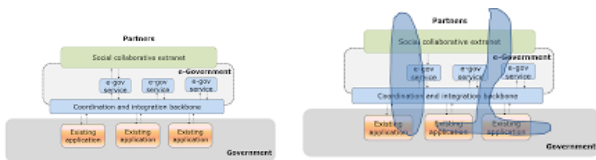


Fig. 14. Application architecture – introductory stage

As existing applications are evolving, they will be replaced by processes and services as well thus creating transitional application architecture as shown in figure 15.

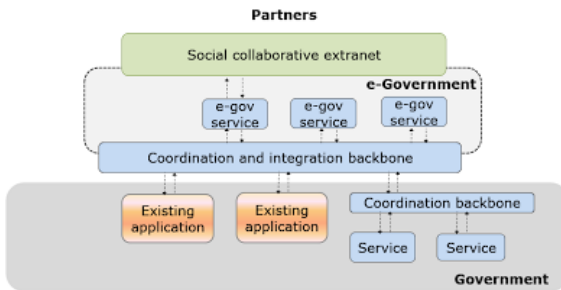


Fig. 15. Application architecture – transitional stage

With converting all previously monolith applications into processes and services, the target application architecture will be reach as show in figure 16. E-government applications will be just connections to a cloud of governmental services.

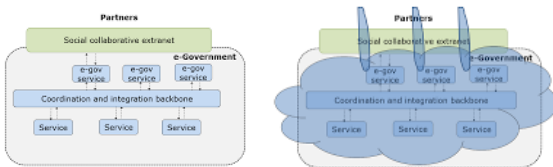


Fig. 16. Application architecture – target stage

And, moving e-government to the really e-social system, the application architecture will morph into a social cloud interacting with the government service cloud as shown in figure 17. The latter serves as a platform for social, professional, private, voluntary and other services to be integrated into e-social system.

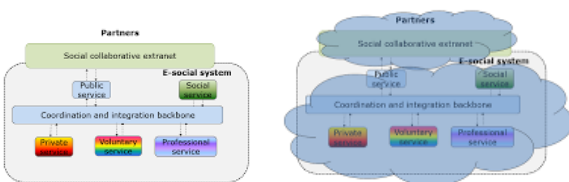


Fig. 17. Application architecture – e-social system stage

For existing e-government systems the evolution from the introductory architecture upwards does require primarily the systematic use of BPM and SOA. Green-field e-government initiatives may start from the target architecture.

All stages form a sort of ladder for step-by-step evolution as shown in figure 18.

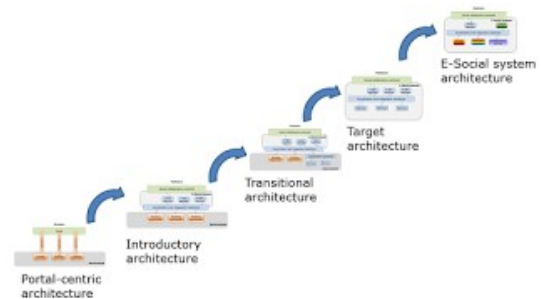


Fig. 18. Application architecture – all stages as a ladder

VI. USE OF SOA AND BPM

The application architecture should be based on BPM and SOA. From the business point of view on BPM/SOA, an executable process coordinates some services as shown in figure 19.

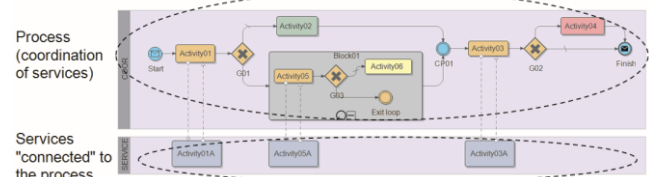


Fig. 19. BPM and SOA are working together to have executable processes

But from the IT point of view, there may be many different services around each process, because a process requires various artefacts: events, roles, rules, documents, data, audit trails, and performance indicators. To help structure different services and other artefacts around processes, a multi-layer implementation model (see figure 20) is proposed for BPM/SOA solutions. In this model, each layer is a level of abstraction of the business and addresses some particular concerns.

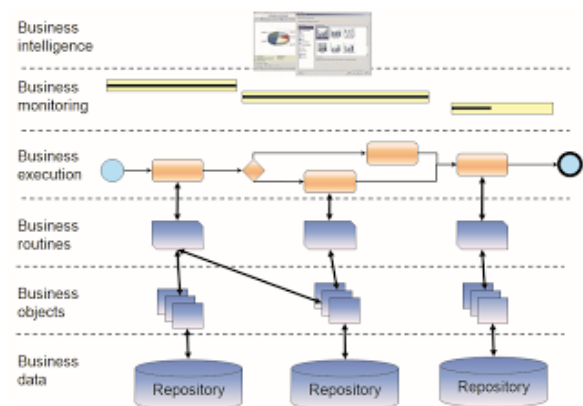


Fig. 20. Multi-layer implementation model for BPM/SOA solutions

VII. APPLYING THE E-GOVERMENT REFERENCE MODEL

Imagine, that a country wants to implement its e-government and e-governance correctly. One of the critical

measures of implementation "correctness" is minimal (ideally, total avoided) duplications in its e-government and e-governance implementations while following the goals and priorities of the country.

Typical country has many governmental entities which are spread in the following levels of government:

- National (or federal);
- Federal ministries and agencies;
- Regional (or cantonal or provincial) authorities with their ministries and agencies;
- Districts (and sometimes sub-districts) authorities, and
- Municipal authorities.

From the government-as-a-system point of view, there are several communicating governmental entities which are working together as one whole to provide services to citizens, business and other non-governmental/social organisations. Without any doubts, e-government and e-governance is a complex system:

- Unlimited life-cycle (unpredictable and incremental evolution).
- Socio-technical system (**how** you do something is sometimes more important than **what** you do).
- Collaborative system.
- Industrialised system.
- Ability for rapid innovation is important.
- Variety of services (several hundred governmental services are listed in the Swiss e-government catalogue [1]).
- High level of security for personal data.

A system of such complexity must be properly architected with all power of enterprise architecture and other technologies to avoid duplications and reworks. So how many enterprise architecture projects should be carried out within the country?

One estimation is "Each governmental entity within the country will be required to define their enterprise architecture for provisioning of e-Governance Services and improving the efficiency and effectiveness of the public sector." If the Country there are 40 ministers, 18 provinces, about 100 districts (with a few sub-districts each) and unknown number of communes then more than 200 enterprise architecture projects are necessary.

Another estimation is based on the e-government reference model and it analyses variations between each level.

A. Municipal (and district) level

By definition, all communes and districts provide the same services. Thus the technical, data, application and business architectures should be the same. Although those governmental entities may have different capabilities, the enterprise pattern

"Asynchronously decentralized organization" [2] provides guidance how to handle this situation.

B. Provincial level

All provinces provide the same services, but with some internal variations as their structures, priorities, even laws may be different. Nevertheless, the technical, data, and application architectures should be the same. At the business level, different provinces carry out the same e-government and e-governance services via slightly different processes which use the same services. The blogpost [3] shows how to achieve a common set of services and assemble them into your slightly different processes.

Important, that the modelling of business processes must be standardised so different business analysts find similar services in similar capabilities. An example of such a modelling procedure is described in [4].

C. Ministerial level

As different ministries have different core businesses, their architectures will be different in some extent.

- Technical architecture can be almost the same (except if some unique functionality is required).
- Data architecture of each core business actually must be shared among the whole e-government and e-governance thus to be aligned at the national level.
- Application architecture of each ministry consists of 20-40 % of unique applications and the rest are common (within the whole e-government) applications. If possible, all applications (the both unique and common) are architected in a similar way with the use of BPM and SOA.
- Business architecture may look very different unless processes patterns are systematically used. The latter act as common building blocks for constructing unique business processes. Various process patterns are available at [5]. Being armed with those patterns, the governmental entities will be concentrating on the unique business challenges and not wasting time for re-inventing the wheel.

D. National level

This level is, practically, a decision-making mechanism which works together with the ministerial level. The vast majority of work is to prepare, validate and approve documents in accordance with several decision-taking patterns (see [6]).

E. What is common?

Technical and data architectures are practically common. Application architecture is shared at about 80-90 %. Business architecture is shared at about 70 %.

With this amount of similarities (see table I), there is only one enterprise architecture project is necessary. Such a project must be carried out on the national level. Other projects are just solution-architecture projects to develop common components.

TABLE I. SIMILARITIES BETWEEN LEVELS OF GOVERNMENT

	Technical arch.	Data arch.	Appl. arch.	Biz arch.	# EA projects
Communal	100 %	100 %	100 %	100 %	0
Provincial	100 %	100 %	100 %	80 %	0
Ministerial	90%	100 %	60-80 %	70 %	0
National	90 %	100 %	70 %	50 %	1

VIII. CONCLUSIONS

The e-government reference architecture is the way to bring the conceptual integrity. The latter is mandatory to reach some coherence in decisions about:

- Optimisations for users (external and internal).
- Compliance with laws.
- Avoiding duplication of work among ministers, provinces, municipalities.
- Simplification of evolution of applications.
- Realisation of the national strategy.
- Natural evolution of e-government.
- Standardization of business processes fragments.

- Unification of software.
- Introduction of new tools and technologies.

As shown in the paper, the consistent use of BPM, SOA and ECM allows addressing various e-government concerns and removing many duplications in a systemic, technology-independent manner within the whole country and, potentially, among several countries.

REFERENCES

- [1] <http://www.ech.ch/vechweb/page?p=dossier&documentNumber=eCH-0070&documentVersion=4.00G>
- [2] <http://improving-bpm-systems.blogspot.ch/2013/09/enterprise-patterns-anisotropically.html>
- [3] <http://improving-bpm-systems.blogspot.ch/2013/02/enterprise-patterns-peas-example-and.html>
- [4] <http://improving-bpm-systems.blogspot.ch/2013/07/bpm-for-business-analysist-modelling.html>
- [5] <http://improving-bpm-systems.blogspot.ch/search/label/practical%20process%20patterns>
- [6] <http://improving-bpm-systems.blogspot.ch/2013/10/practical-process-patterns-decision-as.html>