Automatic Generation of Roadmap for e-Government Implementation

Mauricio Solar, Daniel Pantoja and Gonzalo Valdés

Universidad Técnica Federico Santa María (UTFSM), Chile
{msolar, daniel.pantoja, gvaldes}@inf.utfsm.cl

Abstract. Evaluating readiness of individual public agencies to execute specific e-Government programs and directives is a key ingredient for wider e-Government deployment and success. This article describes how the e-Government Maturity Model (eGov-MM) identifies specific areas in which public agencies should focus improvement efforts. eGov-MM is a capability maturity model, that identifies capability levels for each critical variable and Key Domain Areas, proposes a synthetic maturity level for institutions, and automatically generates the roadmap for each evaluated public agency. In this article, the automatic generation of the proposed roadmap is detailed.

Keywords: e-government, roadmap, capability, maturity model.

1 Introduction

The eGov-MM model (e-Government Maturity Model) [1] is a model to measure public institutions readiness to manage and implement e-Government. It considered several information sources [2, 3, 4, 5, 6, 7] in its initial formulation summarized in [8]. The model was piloted with seven (initially 9) public agencies, and a tuned version was generated which incorporates the participants' feedback and an eGov implementation *roadmap* for each evaluated public agency. An associated self-assessment Web tool was also built and similarly validated [9].

eGov-MM is not only a diagnostic tool, but also a generator of improvement *roadmaps*. The model proposes concrete *roadmaps* for capability improvement, i.e. directives about where the organization financial and human resources should be canalized to improve its capability to carry out eGov initiatives. This *roadmap* is automatically generated.

Section 2 describes the eGov-MM, its main characteristics, objectives and how an organization maturity is evaluated; Section 3 explains the design of the automatic generator of *roadmaps*; Section 4 presents the results and analysis; and Section 5 summarizes and concludes.

2 Model of Capabilities and Maturity

One goal of the eGov-MM is to generate *roadmaps* for progressive evolution of capabilities and maturity. To this end, first we describe the generic model used to define the capability levels for each model variable; then we describe the relation between variables capabilities and their respective Key Domain Areas (KDAs); and finally we describe how the organization maturity is determined from its KDA capabilities.

2.1 Capability and Maturity Levels

The capability is a measurement of the state of each KDA that contributes to support the organization development. The capability of a KDA is determined using the capability level (*CL*) of each of its Critical Variables, i.e. what is really evaluated is the capability of these variables to satisfy certain requirements. The capabilities of the critical variables are weighed according to their importance to give a final KDA *CL*.

The *CL* of a KDA and its critical variables determine the organizational maturity level (*ML*). The *ML* is a property of the organization as a whole; each *ML* corresponds to a predetermined configuration of KDAs in predefined *CLs*. The model allows, once the current *ML* is assessed, to identify the states required to advance to a higher level and propose a "*roadmap*" to improve the organization.

2.2 Generic Capabilities Model

For each KDA there is a measurement scale from 1 to 5, associated with a generic qualitative capabilities model that ranges from "*initial capabilities*" to "*integrated capabilities*"; the values are shown in Table 1.

Table 1. Capacity levels of the maturity model

Capacity Level	Description
Level 1: Initial capabilities	There is evidence that KDA has been recognized and needs to be approached
Level 2: Developing capabilities (repeatable but intuitive)	There is no formal training or procedures spreading, and the responsibility to follow them falls to each individual
Level 3: Defined capabilities	The procedures are not sophisticated, but are the formalization of existing practices
Level 4: Managed capabilities	Established standards and norms are applied through the organization. Tools are mainly automated.
Level 5: Integrated capabilities (Optimized)	Procedures have become best practices, and continuous improvement is applied

The capability levels of each KDA are built from the levels of their variables. For each level, several aspects are considered (incrementally in each level): awareness; human resources training; communication; procedures and practices; compliance of standards and norms; tools and automation; and involvement and responsibility.

2.3 Relationship between Capability Variables and KDAs

The capability level (*CL*) of a KDA is generally the average of the *CLs* of its critical Variables V_i. To accommodate eGov strategies or country development levels with different variables relevance, weights are used for each variable group. Thus, the *CL* of a KDA is the *weighted average* of the *CLs* of its variables V_i (see Eq. 1).

$$CL_{KDA} = \frac{\sum_{i=1}^{n} CL(V_i) * W_i}{100} \quad \text{(Eq. 1)}$$

The weights W_i used in the first model applications (pilot and initial massive application) are shown in [9].

2.4 Organizational Maturity Model

Maturity is a property of the organizational unit as a whole, and the maturity level (*ML*) is obtained from the KDA capacity levels that the unit has. Thus, each *ML*:

- Frames a set of KDAs for a given *CL*.
- Establishes equivalence among eGov implementation maturity of units.

There are several options to determine an organization maturity, namely:

1. Minimum *CL* among all KDAs
2. Average *CL* of all KDAs
3. Predetermined KDA configuration, using a set of values for all KDAs in model.
4. Configuration of high-priority KDAs, using a set of *minimum* values for all KDAs in the model.

The last criterion (Configuration of high-priority KDAs) was adopted in eGov-MM [9]. The organization *ML* is determined (Eq. 2) by a set of values for all KDAs in the model.

$$ML_1 = \text{Conf}_1(CL(\text{KDA}_1), \ldots, CL(\text{KDA}_i)) \quad \text{(Eq. 2)}$$
$$ML_2 = \text{Conf}_2(CL(\text{KDA}_1), \ldots, CL(\text{KDA}_j))$$
$$\ldots$$
$$ML_5 = \text{Conf}_5(CL(\text{KDA}_1), \ldots, CL(\text{KDA}_k))$$

This mechanism was selected for eGov-MM for its flexibility to allow graduating progress according to specific eGov strategies, since it only requires to fixing a minimum set of KDAs that are important for a given *ML*; development criteria and rates for other KDAs are left to the organization. The actual criteria to use can be extracted from domain specialists or agencies leaderships; e.g. from phrases such as:

- "Maintaining enterprise architecture is an advanced issue, which allows aligning business objectives and computer networks … and thus should not be requiring nor evaluating for lower *MLs*…"
- "It is very important to start by aligning the IT, eGov and of business strategies … this should be required even for lower *MLs*".

3 Design of the Automatic Generator of *Roadmaps*

Roadmaps play a leading role in the continuous improvement cycle proposed by the model, since they constitute the direct recommendations to advance to a higher level of maturity.

It is therefore, fundamental for the design of a *roadmaps* system to consider the challenge it implies to automatically generate recommendations, and in particular for this model, it has a high complexity standard since the system must consider strategic aspects that will indorse an optimum upgrading through the maturity levels (*ML*).

3.1 Previous considerations

To guarantee that the *roadmaps* automatically generated by a system are optimum, 4 key factors must be considered: bounding KDAs; variables weight; load balance; and "almost" variables, hereunder detailed.

Bounding KDA

The first factor is to generate *roadmaps* for those KDAs that are bounding the upgrading to the next level of maturity. This means to only generate *roadmaps* of the areas that do not accomplish with the *CL* necessary for the next *ML*. This consideration is because in most of the cases there will be *CLs* of the KDAs that will meet with the requirements of the next *ML* and only a few that are limiting the advance.

Variables Weight (W_i)

This is a high relevance factor, since the fundamental basis of the quality or the optimization of the *roadmaps* system lies in the weight of the variables, W_i.

Considering that the variables have a weight W_i that represent their importance within the KDA, it is fundamental to mathematically control the supply to the *CL* of the KDA represented by the increase of the *CL* of that variable.

In most of the cases, following this logic, it will be only necessary to increase the *CL* of an amount of variables less than the total quantity of the area in order to obtain the increase of a *CL* of the total KDA.

Load Balance

This factor establishes that *roadmaps* must generate an advance with homogeneity along the *CLs* of the variables of the institutions' KDAs. The basis of this factor is a contradictory concept in the short term, but fundamental in the long term, since the maturity levels implicitly involve a homogeneous advance through the *CLs* of the KDAs, and furthermore, in practice the differences between *CLs* of the same area variables create withdrawing behaviors from the optimum management.

"Almost" Variables

Another fundamental factor that must be taken in consideration in the system design are the "almost" variables that allow a variable – in spite of having the same *CLs* of the *roadmaps*. These variables classification correspond to those that although they are in a certain *CL*, they meet with the conditions required of a higher *CL*. This is a common case since in spite of the quality of the *CL* classification by variable, reality is always more complex, and another can be closer to upgrading to the next level.

In the design context of the *roadmaps* system, these variables must be favored in fair conditions with respect to other variables, since its increase of *CL* will require less effort that a common variable.

3.2 Algorithm Logic of the Automatic *Roadmaps*

Automatic *roadmaps* generation algorithm logic developed considers all the key factors above mentioned to assure that the generated *roadmaps* are the best recommendations so the institution can accomplish the next *ML*. This logic is divided basically in the following 5 main stages.

3.2.1 Determination of Limiting KDAs

This phase is based on one of the 4 key factors to be considered, which postulates that in most of the cases the institution can be limited to upgrade to the next ML for only a few KDAs and not for the total of them.

Therefore, the first step of the algorithm consists in identifying the KDAs that are limiting the upgrading, so as to later only generating *roadmaps* in these areas.

In case none of the KDAs of the institution meet with the *CL* of the following *ML* all the KDAs are considered limiting KDAs.

3.2.2 Definition of the Levels that the Limiting KDA must increase

This step is in charge of the immediate management of a technical problem, that is to say, that the limiting KDAs could need to increase more than one *CL* so the organization will increase only one *ML*. Because of this, the second stage of the algorithm consists in reviewing how many *CLs* must increase each area so as to allow the upgrading of the institutions to the next *ML*.

However, the *CL* of the KDA is obtained by means of a weighting of the *CLs* of the variables (Eq. 1), which eliminates the decimal component of the final result. Therefore, in most of the situations there is a remainder of the percentage of the *CL* of the area that must be considered to optimize the *roadmaps* generation.

For example, Table 2 shows a KDA with its corresponding *CLs*.

Table 2. Capability level of variables into a KDA

KDA	V_i	W_i	*CL*
Vision, Strategies and Policies	Communication to stakeholders	25	2
	Strategies Alignment	15	1
	High Management Commitment	30	1
	Resources Asignment for eGov	30	1

Applying equation 1 for the calculation of the *CL* of the KDA, the results of Eq. 3 shows that the KDA "*Vision, Strategies and Policies*" is in a *CL* 1, but has a 25% capability from the second level, therefore, *roadmaps* are optimum, they just must increase the *CL* of the area in a 75% and not in a 100%.

$$CL_{KDA} = \frac{50+15+30+30}{100} = \frac{125}{100} = 1.25 \qquad \text{(Eq. 3)}$$

3.2.3 Identification of the "almost" Variables by Area

This step of the algorithm is directly related to one of the key factors mentioned before, "almost" variables. It is known that these variables are of great importance within the *roadmaps* generation, positions that will be preferential when generating them, since the increase of a *CL* from an "almost" variable requires less effort than a normal one needs. Specifically in this step each variable of a KDA is analyzed so as to recognize if it is an "almost" variable or not; also, a record is saved with the total amount of "almost" variables of the KDA.

3.2.4 Classification of the Problem of the *Roadmaps* Generation

This step is one of the most important of the *roadmaps* generation algorithm, since it is in this step where the logic to be used when generating the *roadmaps* of each KDA in an optimum manner is decided.

The classification of the problems is carried out based on two parameters: levels quantity that the KDA must upgrade to increase the *ML* of the institution and the amount of "almost" variables contained in Table 3.

Table 3. Problems classification of the *roadmaps* generation

Increase of the *CL*	"Almost" Variables
1	All
1	Non
1	Some
More than 1	All
More than 1	Non
More than 1	Some

3.2.5 Calculation of the *Roadmaps* according to the Type of Problem

The last stage of the *roadmaps* algorithm consists basically in the *roadmaps* generation according to the logic associated to the type of problem defined in the preceding step, which was classified at the same time, as per the information obtained from the first steps.

Although there are 6 classifications of different problems, it has been decided that only 4 logics of *roadmaps* generation are enough for the solution of all the cases; the 4 logics associated to the types of problems are hereunder detailed:

- **One *CL* and only "almost" variables:** Solves the homonymous problem; the characteristics of this problem assure that the increase of the *CL* required will be achieved in the worst of the cases, increasing all the variables of the KDA.

Since all the variables are essentially "almost" type, the increase of the *CL* in any of them require the same effort, therefore, as from this point of view no variables priority exist.

The logic that solves the problem consists of, in first instance identifying the best *CL* contribution of the KDA generated by the increase of a *CL* of one (or more than one) variable that together with the remaining of the *CL* of the KDA could achieve to carry into effect the KDA to the next level. Once detecting what variables will be increased of level, the optimum *roadmaps* will be those that will promote the development of the characteristics of the next *CL* of these variables.

- **One *CL* and none "almost" variable:** Solves the homonymous problem, the characteristics of the problem establish that no priority exists between additional variables to the contribution that each one of them provides to the increasing of the *CL* of the KDA.

  The challenge of this problem lies in the fact that no guarantee exists that all the variables can go through the next level (one of those can be in the maximum level), therefore the algorithm must be able to find the best combination of variables (that will produce the highest contribution to the *CL* of the KDA), and *CLs* that each one must increase, so together with the remains of the *CL* of the KDA they will be able to upgrade the KDA to the next level. The optimum *roadmaps* will be assigned to promote the development of the next *CL* of these variables.

- **One or more CLs and some "almost" variables:** Solves the problems "*one CL and an 'almost' variable*" and "*more than one level and some 'almost' variable*". The reason by which a same logic involves these two problems is that to the principal reasoning the amount of *CLs* that the KDA must increase is indifferent because, as the preceding case, it is not possible to assure that all the variables of the KDA could increase the level (any of them could be in its maximum level) so the algorithm must be able to find the best combination of variables (that will produce the biggest contribution to the *CL* of the KDA) and *CLs* that each one must increase, so together with the remaining of the *CL* of the KDA they can achieve to upgrade to the next level the KDA.

  Unlike the preceding case, in this one it an additional priority to the contribution that each variable performs exists, and it is present in the "almost" variables, since the increase of levels of these variables, require less demand.

  Therefore, the process of generating the *roadmaps* must choose the variables (favoring the "almost" variables and the weight balancing) and the amount of *CL* that each one must increase, that perform the best contribution to the *CL* of the KDA.

- **More than one *CL*, and none or all of the "almost" variables:** Solves the problem "*more than one CL and not any 'almost' variable*" and "*more than one level, and only 'almost' variables*". The reason by which a same logic involves the two problems is that the characteristics of these two problems are similar, in both cases all of the variables have the same priority (except for the contribution) and the fact that all the variables can upgrade their *CL* is not guaranteed (any of them can be found in the maximum level). For this reason the *roadmaps* algorithm must be able to find the best combinations of variables (those that will produce the highest contribution to the *CL* of the KDA), and *CLs* that each one must increase, so together with the remaining *CL* of the KDA they can move the KDA to the next level. The optimum *roadmaps* will have to promote the development of the established *CL* characteristics that must be achieved by these variables.

3.3 Implementation of the Automatic *Roadmaps* Algorithm

Implementation of automatic *roadmaps* generation algorithm has been integrated to the web system of the evaluation and diagnosis of the maturity model and ICT use capability in public agencies. In this way, *roadmaps* are instantly established when the questions associated to each variable are answered.

```
Central Algorithm of the Roadmap
{
 bounding_areas ← areas that bound the next ML
 For each (bounding_areas)
 {
   CL-KDA ← CL of a KDA
   cl ← Levels to be incremented in the KDA
   V_area[] ← List of variables in a KDA sorted by descending weight
   CL_dec ← CL not rounded of the KDA (decimal)
   IF (cl == 1)
   {
    For each V_area[]
      IF (verify_almost(V_area) == true)
      almost_var[] ← V_area
    IF (count(almost_var[]) == count(V_area[]))
    roadmap ← best_contr_all_almost(V_area[], CL_dec);
    IF (count(almost_var[]) == 0)
    roadmap ← best_contrib_zero_almost(V_area [],CL_dec);
    IF (0 < count(almost_var[]) < count(V_area[]))
    roadmap ← best_contr_some_almost(V_area[],CL_dec, almost_var[])
   }
   IF (cl > 1)
   {
    For each V_area[]
      IF (verify_almost(V_area) == true)
      almost_var[] ← V_area
    IF (count(almost_var[])==count(V_area[]))OR(count(almost_var[])==0)
    roadmap ← best_contrib_cero_all_almost_2(V_area[],CL_dec, c_level)
    IF (0 < count(almost_var[]) < count(V_area[]))
    roadmap ← best_contr_some_almost(V_area[],CL_dec, almost_var[], cl)
   }
 print(roadmap)
}
```

Automatic calculation of the *roadmaps* is divided in two parts: a central algorithm in charge of the general management and a specific function dedicated to each of the 4 logics described above. Above you can see the central algorithm for the *roadmap* and below one of the 4 functions assigned to manage each of the logics.

```
function best_contr_all_almost(V_area, CL_dec)
{
 delta_capacity_necessary ← (floor(CL_dec)+1)-CL_dec
 delta_comparative ← 0
 levels_to_rise ← 1
 for each V_area
 {
   IF (V_area → level_capacity_question < 5)
   {
     delta_comparative += V_area → weight/100
     roadmap ← array(levels_to_rise, V_area)
   }
   IF (delta_comparative > delta_capacity_necessary)
   return roadmap
 }
 return 0
}
```

4 Results and Analysis

The model was applied to a small set of public agencies as a validation mechanism, but in this section we present the results of applying the eGov-MM and its automatic *roadmap* generator tool to one public agency; it describes the main sample characteristics, the results of capability measurement for the sample, and the generated *roadmaps*.

The *ML* obtained by the reported agency was the minimum level (Level 1). This reality priori disagrees with what was observed and expected, so, in order to understand the result obtained, a deeper analysis should be carried on.

_CL_ by Area

The *CLs* of the areas are found mainly in the two first levels. There are areas with a higher *CL* (level 3) that is worth to mention: “Performance management”, “Infrastructure and eGov Tools”, and “Change management”.

These higher capacity levels are mainly possible because the agency has demonstrated that a suitable infrastructure was prepared in order to improve the quality of their services, and that an adequate management and control of the functionaries and of the work they carry on was also considered.

Furthermore, it is remarkable one area that is in a high *CL* (level 4), “Inter-operability Practices”. This area reflects the agency understanding about the importance of working with other agencies in an integrated manner, and they have also completely assumed the challenge that interoperability brings both, at the institutional and technical level.

CLs by Variable

At variables level reality is similar; the *CLs* are mainly distributed in the two first levels. Those areas that accomplish a higher *CL* are homogeneously supported by the *CL* of their variables.

It is interesting the case of 3 variables that accomplish the maximum *CL*: "Basis Infrastructure Hardware/Software", "Institutional Interoperability" and "Customers' satisfaction".

In outline it can be observed that the public agency has prepared an adequate interoperability platform to their initiatives of electronic government.

They fully meet with their task of knowing the necessities and requirements of the citizens and have disclosed the importance of counting with interoperability services so as to be related to other agencies and improving the quality and access easiness to the services that the agency offers.

Roadmaps

As above was mentioned, the calculation of the *ML* depends on a pre-established definition of the *CLs* by area for each real level and the proximity of the following *ML*.

The web system of evaluation and diagnosis of the Maturity Model and Capability of the ICT use in public agencies points out that the organization is in level 1 for 3 KDAs, that is to say, there are 14 KDAs that meet with the *ML* 2, and only 3 that do not.

The 3 KDAs that are bounding the advance to the next *ML* are: "IT Management and Organization", "Regulations Compliance", and "Human Capital".

Also, 2 of these 3 bounding areas are very close to upgrading their *CL*, so moving the agency to the next *ML* is a task almost easy to carry on. Considering that it is only limited by 3 areas, and because 2 of the 3 areas have variables that are at just one step of upgrading to the next *CL*, in consequence, the area is very close to its upgrading to the next *CL*.

Roadmaps automatically generated by the system indicate that for the institution to accomplish *ML* 2, it has only to meet with the requirements of Table 4.

Table 4: *Roadmap* automatically generated

IT Management and Organization (amount of levels to increase: 1)
IT Infrastructure Planning (amount of levels to increase: 1) • IT Unit is incorporating the IT Infrastructure Planning activities but only to a tactical level • IT Unit has developed a documented Infrastructure Plan but it is not consistently applied • Staff of the IT Unit have accomplished the planning skills but not in a structured way
Regulations Compliance (amount of levels to increase: 1)
Internal Regulation (amount of levels to increase: 1) • The High Board already defined or is analyzing the mechanisms to be endowed of a diffusion strategy of the internal regulations, to facilitate the eGov initiatives.
Human Capital (amount of levels to increase: 1)
Management of eGov Competences (amount of levels to increase: 1) • The model that allows a Management based on the Competences integrates the competences definitions that the eGov requires for all the roles.

Regarding the agency, the conclusion is that, although it obtained a minimum *ML* it is very close to accomplishing the next level. Also, according to the *CLs* of the variables and areas obtained by the agency, it is observed that an adequate interoperability platform exists for the development of the ICT use, besides, there is an excellent identification of the citizens' needs, and the new technological developments receive from the high commands the basic minimum necessary support. However, the vision and strategy of the ICT use is limited, no clear planning of the path to follow in the use of them exists. Therefore, generated *roadmap* is in accord with the institution's reality.

5 Conclusions

The information provided by the Maturity Model is useful in taking decisions with respect to the ICT strategy and direction, and is a feedback tool for the institution management which, as a consequence allows the generation of a continuous improving cycle when discovering the areas that are developed according to the institutions' reality and those that do not.

The model not only performs this measuring, but also generates the *roadmaps* necessary to follow, so as to accomplish the next maturity level in the best way. These are automatically generated as soon as changes in measuring are carried on, so it is assured that no efforts or resources are wasted in variables of areas that do not have an impact in the short or medium term within the organization.

Roadmaps algorithm was developed to be adapted to any situation that could show up when applying the model to the organization; it goes beyond the detail level of variable to consider also the characteristics of each variable.

In this way it is guaranteed that the generated recommendations are the best in relation to the minimum effort to obtain the maximum level, both in capacity and maturity, but furthermore it should guarantee that the institution will develop itself as a balanced organization in the three areas.

The algorithm works correctly and it is integrated to the application of evaluation and diagnosis of the Maturity Model and Capability of the ICT use in public agencies.

As a future work, it would be interesting to add the possibility of addressing the focus to the *roadmaps* generation, so as to allow that the upgrade through the maturity levels do coincide with short term goals. In this way it would be possible that the model will not only lead institution to a higher maturity level through the shortest path, but will also do it through the shortest path that will meet with the short or medium term targets of the IT strategies put into action by the institution.

Thus, eGov-MM is not only a diagnostic tool, but also a generator of improvement *roadmaps*. Government has a methodological and technological tool to measure status and improvement of eGov implementation by specific public agencies.

References

1. Iribarren, M., Concha, G., Valdes, G., Solar, M., Villarroel, M., Gutiérrez, P., and Vásquez, A.: Capability Maturity Framework for e-Government: A Multi-dimensional Model and Assessing Tool. in M.A. Wimmer, H.J Scholl and E. Ferro (Eds.): EGOV 2008, *LNCS* vol. 5184, 136-147, Springer-Verlag Berlin Heidelberg, (2008).
2. Andersen, K.V., Henriksen, Z.H.: E-government Maturity Models: Extension of the Layne and Lee Model. *Government Information Quarterly* 23, 236-248, (2006).

3. Cresswell, A., Pardo, T., Canestarro, D.: Digital Capability Assessment for e-Government: A Multidimensional Approach. *LNCS* vol. 4084, pp. 293-304, (2006).
4. Layne, K., Lee J.: Developing Fully Functional e-Government: A Four Stage Model. *Government Information Quarterly* 18, 122-136, (2001).
5. Wimmer, M.A., Tambouris, E.: Online One-Stop Government: A Working Framework and Requirements. *Proc. of the 17th IFIP World Computer Congress*, 117-130. Kluwer Academic Publishers, Boston (2002).
6. Esteves, J., Joseph, R.: A Comprehensive Framework for the Assessment of e-Government Projects. *Government Information Quarterly* 25, 118-132, (2008).
7. Al-Khatib, H.: A Citizen Oriented E-government Maturity Model, Brunel University, Londres, (2009).
8. Valdes, G., Iribarren, M., Concha, G., Solar, M., Visconti, M., Astudillo, H., Villarroel, M., Gutiérrez, P. and Vásquez, A.: Identifying relevant National e-Government Implementations for an Emerging Country: A Selective Survey, in E. Ferro, H.J. Scholl, M.A. Wimmer (Eds.): *Proc. of the 7th Int. Conf. on Electronic Government*, EGOV 2008, Trauner Verlag Druck Publisher, Torino, Italy, 141-149, (2008).
9. Solar, M., Astudillo, H., Valdes, G., Iribarren, M. and Concha, G.: Identifying Weaknesses for Chilean e-Government Implementation in Public Agencies with Maturity Model, in M.A. Wimmer et al. (Eds.): EGOV 2009, *LNCS* vol. 5693, 151-162, Springer-Verlag Berlin Heidelberg, (2009).