Leveraging Reuse-Related Maturity Issues for Achieving Higher Maturity and Capability Levels

Luigi Buglione[1,2], Giuseppe Lami[3], Christiane Gresse von Wangenheim[4], Fergal Mc Caffery[5], and Jean Carlo Rossa Hauck[4]

[1] Engineering.IT SpA - Via R. Morandi 32, 00148 Rome, Italy
[2] Ecole de Technologie Superieure (ETS) – Montréal, Canada
[3] ISTI-CNR (Italian National Research Council) – Pisa, Italy
[4] Federal University of Santa Catarina (UFSC), Brazil
[5] Regulated Software Research Group & Lero - Dundalk Institute of Technology, Ireland
luigi.buglione@eng.it, giuseppe.lami@isti.cnr.it,
{gresse,jeanhauck}@gmail.com, fergal.mccaffery@dkit.ie

Abstract. During the past 20 years *Maturity & Capability Models* (MCMs) become a *buzzword* in the ICT world. Since the initial Crosby's idea in 1979, plenty of models have been created in the Software & Systems Engineering domains, addressing various perspectives. By analyzing the content of the Process Reference Models (PRM) in many of them, it can be noticed that reuse-related issues have unfortunately often little importance in the appraisals of the capabilities of software organizations while in practice they are considered as significant contributors in traditional process and organizational performance appraisals. While MCMs represent a good mean for assessing the status of a set of processes, integrating two or more models with a common area of focus can offer more information and value for an organization. The aim of this paper is to present some information about Reuse best practices and models, keep the best components from each model and – using the **LEGO** (**L**iving **E**n**G**ineering pr**O**cess) approach to process improvement - merge those best practices from several types of maturity models into an organizational Business Process Model (BPM) in order to achieve in an easier and faster way higher organizational maturity and capability levels.

Keywords: Maturity & Capability Models (MCM), CMMI, SPICE (ISO/IEC 15504), Reuse-related issues, Improvement, LEGO approach.

1 Introduction

Recently somebody said that the 'copy & paste' computer function was one of the greatest inventions of last forty years[1]. It seems just something for kidding, but for instance from a human-computer interaction (HCI) viewpoint it was a very common-sense metaphor from every day reuses practice: copy-paste-edit, moving what yet

[1] http://goo.gl/d3hEo

J. Favaro and M. Morisio (Eds.): ICSR 2013, LNCS 7925, pp. 343–355, 2013.

exists. In terms of estimation practices, it'd lead to analogous estimation. But differently from other practices, reuse was not elevated in the Software Engineering studies and guides to the 'management' level, as it was something 'minor', while it's a fundamental practice and way to manage and plan e.g. product lines. Moreover, from a software measurement perspective, for measurers applying a functional size measurement (FSM) method such as IFPUG Function Point Analysis (FPA), originally reuse was included into one of the so-called GSC (General System Characteristics) in the VAF (Value Adjustment Factor), classified within Non-Functional Requirements (NFRs) and rated with a lower value than the so-called FURs (Functional User Requirement), simply contributing to 'adjusting' the unadjusted FP value (UFP). And being something within the NFR side, it was (and still is) more difficult to evaluate and rate it, also from the process side.

Observing the plenty of 'maturity models' appearing on the ICT arena during last 30 years, there were several ones in well-defined domains such as Project Management (e.g. PM3O or PMMM) or Test Management (such as TMMI or TPI), but few ones about Reuse[2]. This just because 'reuse' is a keyword for a very wide area of action, including – just to name a few – product lines, the organization of software factories for thinking and creating 'objects' to be shared and continuously improved and much more.

Thus, there is a huge need for any organization to first reinforce the knowledge and subsequent application of proper reuse practices and processes (in a broader sense, not strictly in the development terms), starting from a 'functional' reuse (reusing complete chunks of logical data and functionalities for creating new functionalities) towards a 'technical reuse' (reusing physical parts of existing logical data and functionalities for creating new functionalities) within ICT organizations [32]. Unfortunately, little efforts have been made to face such a need.

Right now few studies [21][24][25][28][31] analyzed the way for an evolutionary path to reuse, proposing or discussing the idea for 'reuse maturity models', often compliant with the well-known horizontal models such as SPICE (ISO/IEC 15504) [6][7] or CMMI [3][16][17]. But there was no 'breakouts' as well as done in other specific domains such as Project or Test Management, as previously stated. The aim of this paper is to propose a LEGO (**L**iving **E**n**G**ineering pr**O**cess)[3] application for the Reuse management area, matching together different reuse-specific processes using a four-step process, in order to obtain a comprehensive process to be applied in an organization, which could further enable to have on the technical side better estimates (the more and better reused, the lowest the effort to produce a new software solution) and for the economic side higher ROIs.

The paper is organized as follows: Section 2 discusses a possible taxonomy of MCMs by *orthogonality*, in order to better understand the possible intersections among them. Section 3 proposes a series of specific reuse maturity models and frameworks, for extracting any possible element of interest (EoI) for reinforcing a typical reuse engineering (horizontal) process. Section 4 summarizes the LEGO

[2] To be meant as ISO says as the "use of an asset in the solution of different problems" (ISO/IEC 24765:2010 – Systems and Software Engineering Vocabulary).

[3] **LEGO** is a new approach proposed for helping organizations in building and reinforcing their own process models moving from the combination of single items from multiple maturity & capability models (MCM). More details on Section 4.1.

approach, with its main elements and four-step process and shows the deployment of LEGO to the Reuse Management process, joining the ISO/IEC 15504 REU process area with the EoI from the previously examined reuse models/frameworks. Finally, Section 5 provides some conclusions and the next steps for this work.

2 Maturity and Capability Models (MCMs): Representations and Dimensions

Maturity & Capability Models (MCMs) represent a simple, common-sense mechanism for benchmarking entities of interest (EoI) according to established criteria. Typically most of those models are structured using five maturity levels, as well as in a Likert scale (or by the fingers of a hand). The more mature (or capable) a certain organization (or process), the higher the level. The quality of a MCM can be perceived from users if the practices for a model are properly distributed in a way to do not create any step of the 'maturity stairway' too much challenging, but having a regular progression and evolution towards higher levels. The further evolution in MCM was distinguishing 'maturity' and 'capability'. Maturity is referred to an organization, capability to single processes to be run within an organization[4]. A consolidated capability evaluation can be converted to a maturity evaluation (e.g. in CMMI there is the so-called 'equivalent staging' mechanism [1].

2.1 Why Do We Need Choosing a MCM?

This is why from the release of the Sw-CMM in the early '90s, moving from the Crosby's experience [2], plenty of MCM with the same architecture has been proposed over the years, with more than 40 models yet in 2003 when the term "MM-mania" was coined [8]. Since then, new MCM continue to be proposed joining several issues (e.g. Agile Methodologies, Architecture, Reuse, Testing, etc.)[5]. When more MCM are available within a certain application domain of interest, some suggested criteria for choosing the proper MCM to use for process assessment and improvement activities could be to choose the one that has:

- higher number of missing/improvable elements that we would want to include in our Business Process Model (BPM)[6];
- deeper granularity in the definition of processes.

2.2 Coverage and Classification of MCMs

In order to make comparisons and mapping among different MCM, a series of classifications and taxonomies are needed. For instance, MCM are typically classified by their application domains: Software-System Engineering, Security Engineering,

[4] Definitions of organizational maturity and process capability can be found in the ISO/IEC 15504-7 and ISO/IEC 15504-1 respectively.

[5] An updated list of such models is available at: `http://www.semq.eu`.

[6] For BPM it must be intended the whole process management system of an organization, wider than the solely summation of several PRM from distinct maturity models as CMMI.

Usability, etc. Few years ago, we proposed another possible criterion, looking at them in terms of *orthogonality* of the content of their PRM along the project lifecycle [19][20]:

- **Horizontal** - some of the MCM have processes that go through the whole supply chain, from requirements till their delivery: they could be classified as '*horizontal*' models. Examples of horizontal models in the ICT world are CMMI, ISO 12207/15504 or the FAA i-CMM [11].
- **Vertical** - other MCM focus on a single perspective or process category: they could be classified as '*vertical*' models [9], because going into a deeper detail on a specific viewpoint. Examples for the second group includes e.g. TMM [12] or TPI [13] in the Test Management domain, and P3M3 [14] and OPM3 [15] in the Project Management domain.
- **Diagonal** - the third categorization refers to those models whose content is in a middle way between Organizational and Supporting processes, and this is referred here as *diagonal* models. People CMM (P-CMM) [4] is an example for such category.

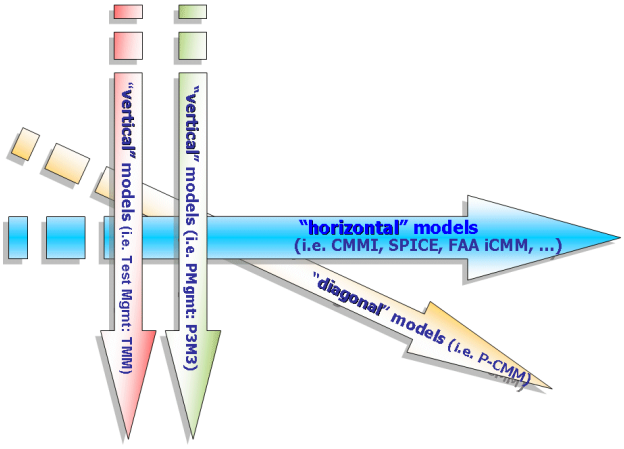

Fig. 1. A classification of Maturity Models

But the final purpose of an organization is to globally improve its BPM, results and performances by higher maturity & capability levels in its practices. And the usage of a single MCM, no matter if quite comprehensive, cannot be the final solution by a mean to achieve the desired outcomes: more MCM should be selected and joined, according to the organization's needs, maturity and capability levels at a certain moment in time. Nonetheless it would be shared thought[7], no practical ways to put it into practice have been suggested right now.

3 Reuse-Related Issues in Typical Horizontal MCMs

A question to pose is: are reuse-related issues adequately considered and evaluated in the overall context of a process improvement initiative with the current MCMs for

[7] E.g. SEI's PRIME (`http://www.sei.cmu.edu/prime/`) initiative or this 2008 SEI's study.

Software & Systems Engineering? Now it's time to take into account two of the most popular MCMs for Software & System Engineering in order to discuss the extent they address reuse-related issues: CMMI-DEV and ISO/IEC 15504 standard. We are moving from the two more known horizontal MCMs (H-MCM) because in such way it's possible to have a value-chain view, using 'reuse' as part of the whole picture and not as a detail to be analyzed separately. While CMMI includes a specific Process Reference Model (PRM), the ISO/IEC 15504 standard does not. In fact, ISO/IEC 15504 gives (in its Part 2) just a set of requirements to define a compliant PRM (i.e. a PRM having the needed characteristics to be applied in the assessment and improvement mechanism the standard itself provides). However, the ISO/IEC 12207 standard [10] provides a PRM for software that has been defined in a compliant way as respect the requirements defined in ISO/IEC 15504. Thus, it is not surprising that in the practice the ISO/IEC 12207 PRM is widely used in the application of the ISO/IEC 15504 standard. For these reasons, in the following of this paper we will use the term ISO15504-12207 to refer to the ISO/IEC 15504 standard for Process Capability Determination and Improvement + the compliant PRM defined in the ISO/IEC 12207 standard. Table 1 presents a summary of some of the reuse-related issues included in those two maturity models. The first evidence is that reuse is not addressed by these two PRM in the same way. In fact, the CMMI does not include any Process Areas directly addressing reuse issues but only a couple of practices in Technical Solution (TS) process area; on the contrary the ISO/IEC12207/15504 provides a process group on reuse composed of three processes (REU processes). Moreover it is possible to observe that reuse-related issues are mostly present as appraisal criteria rather than in terms of single processes capability/maturity indicators in the respective PRM.

Table 1. Reuse-related issues in CMMI-DEV and ISO models

Model	**CMMI-DEV**	**ISO 15504/12207**
Domain	Sw-SE	Sw-SE
PRM (source)	CMMI-DEV v1.3	ISO/IEC 12207
PRM (no. processes)	22	47
Process Categories	4 (Engineering, Process, Project, Support)	9 (Primary: Acquisition, Supply, Operation, Engineering; Organizational: Management, Reuse, Resource & Infrastructure, Process Improvement Management; Support: Supporting)
PRM reuse-related processes	None (Reuse practices are partly dealt with)	3 (REU.1 Domain Engineering; REU.2 Reuse Assets Management; REU.3 Reuse Program Management)
PAM ext. appraisals	SCAMPI v1.3 [5]	ISO/IEC 15504-2 [6] ISO/IEC 15504-5 [7]
PAM reuse-related issues	TS-SP-2.1 (Develop Alternative Solutions and Selection Criteria) TS-SP-2.4 (Perform Make, Buy or Reuse Analyses)	26 REU.1, REU.2, REU.3 related BPs

Starting from the information provided in Table 1, we can discuss in more detail the way CMMI and ISO15504-12207 address reuse-related issues both form the process definition and the appraisal/evaluation side.

While the CMMI, on the process side, does not include any direct reference to reuse-related issues, the ISO15504-12207, because it includes three processes directly addressing reuse, covers the principal aspects reuse implies both from a technical and managerial viewpoint. In particular, the processes included into the Software Reuse Group are: Domain Engineering process (aimed at developing and maintaining domain models, domain architectures and assets for the domain), Reuse Assets Management process (aiming at managing the life of reusable assets from their conception to retirement), and Reuse Program Management (aiming at planning, establishing, managing, controlling, and monitoring the organization's reuse program and systematically exploiting reuse opportunities).

On the appraisal side, CMMI presents the Specific Practice 2.4-3 "Perform Make, Buy, or Reuse Analyses" associated to the Specific Goal SG3 "Implement the Product Design" of the Technical Solutions (TS) Process Area. Also the Specific Practice 2.1-1 "Design the Product or Product Component" associated to the Specific Goal SG1 "Select Product-component Solutions"" of the same Process Area, addresses, but only in an indirect way, reuse.

The 26 Base Practices (i.e. process performance indicators) associated to the three reuse-specific processes represent the way reuse is referred by ISO15504-12207 from the appraisal viewpoint. Nevertheless, because according to the ISO/IEC 15504 scheme the Base Practices cannot be used as Capability Indicators, from a Capability/Maturity perspective such a standards doesn't take into account reuse as an indicator of Process Maturity for the overall software process (to be intended as composed of a sub-set of the processes provided by the PRM).

On the basis of the previous considerations, reuse-related issues aren't adequately considered and evaluated in the overall context of a process improvement initiative according to the two principal MCMs for Software & Systems Engineering considered. Therefore two main possibilities arise for improving the reuse-side of the organization:

- Setting up and managing distinct appraisal initiatives for the different domains of interests (with their related PRM) and after coordinating results for a common, improvement plan within the organizational BPM scope;
- Managing a single appraisal initiative, merging before the process elements into a single PRM.

Thus, we started to explore what was produced during last past 20 years in terms of MCMs on Reuse, summarizing the most relevant information in Table 2.

Table 2. Some Reuse Models/Frameworks

Model/ Framework	Repr. Type	ML (#)	Architect-Type	Comments/Notes
[21] RCMM (Reuse CMM)	Staged	5 [1-5]	Level-based	---
[22] Management tool	Staged	6 [0-5]	Level-based	Series of typical agile reqs verified + 14 BTOPP elements for reusing factors
[23] REBOOT approach (and Reuse Maturity Model)	Continuous	---	---	23 Key process areas in 5 process categories
[26] RMM (Reuse Maturity Model)	Staged	5 [1-5]	Matrix-based	5 MLs, 10 key reuse drivers
[27] 3RMM	Staged	5 [1-5]	Level-based	Several scalability factors and reuse variables
[29] RCM (Reuse Capability Model)	Staged	4 [1-4]	Level-based	4 Critical Success Factor Classes for reuse capability improvement provided
[30] RMM (Reuse Maturity Model)	Staged	6 [1-6]	Level-based	Suggestions for integrating reuse practices within the (old) Sw-CMM
[35] Lim's model	Staged	5 [1-5]	Matrix - based	10 factors of influence (drivers)
[18] RiSE Maturity Model	Staged	5 [1-5]	Matrix-based	Macro-goals for each level; 10 Factors of influence (organizational; business, technological; process)

4 Experiencing LEGO to Reuse Management

4.1 The LEGO Approach

Recently a common-sense approach, called **LEGO** (**L**iving **E**n**G**ineering pr**O**cess) [33] was proposed for stimulating organizations to improve their own processes, taking pieces (such as the real LEGO bricks) from multiple, potential information sources to be integrated to form a unique, reinforced picture for a particular process or set of processes. The starting point – for this paper – is that any model/framework can represent only a part of the observed reality, not all of its possible views, simply because it needs to represent one single viewpoint at a time. Thus, through handling similar elements from different sources, we can hopefully find more 'fresh blood' for improving the organizational processes. LEGO has four main elements, as shown in Figure 2:

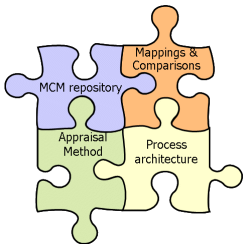

Fig. 2. The four elements of the LEGO approach

1. a 'Maturity & Capability Models' (MCM) repository (www.gqs.ufsc.br/mcm), from relevant processes or MCMs (meaning also the other dimensions – not yet the process dimension) can be identified;
2. knowledge about the process architecture of each model, for understanding how to transform desired elements from a certain model into the target format, especially when considering that the source models may have different architectures that need to be integrated into a single model;
3. mapping(s) & comparisons between relevant models, in order to understand the real differences or the deeper level of detail from 'model A' to import into 'model B';
4. a process appraisal method (PAM) to be applied on the target BPM (Business Process Model).

LEGO has also a related four-step process:

1. **Identify your informative/business goals:** clearly identify your needs, moving from the current BPM version and content.
2. **Query the MCM repository:** browse the MCM repository, setting up the proper filters in order to obtain the desired elements (processes; practices; etc.) to be inserted in the target BPM.
3. **Include the selected element(s) into the target BPM:** include the new element(s) in the proper position in the target BPM (e.g. process group, maturity level, etc.).
4. **Adapt & Adopt the selected element(s):** according to the process architecture of both process models (the target and the source one), the selected elements may need to be adapted, tailoring such elements as needed.

4.2 Applying LEGO to Reuse

One of the main requirements for improving estimates saving time by building more consistent systems is to reinforce the management of reuse practices from an overall viewpoint, from their elicitation through to the day-to-day management, as shown from a long time e.g. by QIP [34].

The focus of this work is exclusively on external models as opposed to actual (living and active) organizational practices, so that any reader can easily access to the original sources and fully understand the LEGO process, that could (eventually, if interested) be replicated in his/her own organization through forward moving from their existing organizational Business Process Model (BPM). Our aim is to show how to hybridize ideas for obtaining a better and more comprehensive final result. Thus, we list the preconditions, process and main results from the application of the LEGO process to the Software Reuse domain, in order to propose a better process that may be applied in an organization:

1. **Identify your informative/business goals:** Improve the estimation capability and results by a refinement in the overall management of requirements (business, technical):
2. **Query the MCM repository:** In this paper we consider a sub-set of the ISO/IEC 12207 reuse-related processes (i.e. belonging to the REU process Group) as the baseline for working upon, adding eventual practices from the other Reuse-related

models/frameworks listed in Table 1. After a detailed analysis, we discarded some of the above presenting models, in particular [21], proposing only a high-level staged path with no detailed elements, focusing on the remaining ones. Table 3 proposes the list of potential elements of interest (EoI) to consider for improving ISO/IEC 12207 reuse-related processes.

Table 3. Reuse Maturity & Capability Models (MCM): Elements of Interest

Model/ Framework	Elements of Interest (EoI)
Management Tool	• The study considers a series of characteristics typical to the Agile Developmernt/Management domain • It considers also from a continuous perspective a Level 0 for 'no reuse' • It considers 15 reuse factors linked to the maturity levels by categories (business; domain; organizational; process; people; technology) • Appendix G presents a mapping between the 15 factors and CMMI-DEV process areas (w/strength) • Appendix H presents a summary of the 15 factors scaled by maturity level (suggestion)
REBOOT Reuse Maturity Model	• Deal with organizational and technical aspects of reuse • 23 Key process areas in 5 categories (organization; Project Management; Dev. Process; Library; Metrics)
RMM	• 10 key drivers considered, using a matrix-based representation/approach • Particularly stressed the people/organization drivers, as well as the legal/contractual issues • To be inserted into a level-based structure
3RMM	• Information on Environments (Repository, Software, Information) + Administrative Management
RCM	• 21 Critical Success Factors corresponding to issues most critical to improving reuse capability • Intended for self-assessment and planning purposes
Lim's model	• Particular attention to the following factors o Motivation/culture; planning for reuse; metrics; legal issues; reuse inventory (assets)
RiSE Maturity Model	• Representation of the influence factors using a matrix-based view, retrieving also from past experiences and models • Particular interest for the following factors: o Organization (Software reuse education, Rewards and Incentives; Independent team) o Business (Product family approach) o Technology (Repository system usage) o Processes (Quality Models, Measurement, Origin of the reused asset)

3. **Include the selected element(s) into the target BPM:** Looking at the analysis of potential EoI in Table 3. The main improvements/suggestions seem to be mainly associated with the REU processes. Table 4 shows how our suggestions were introduced in the current REU processes, describing a new possible improved process that may be mapped against your own QMS internal process(es) covering that subject. The solely REU.1 process was not taken into account because its purpose, having very few details to be improved observing the reuse models listed above in previous table.
4. **Adapt & Adopt the selected element(s):** After adapting the original REU processes, as shown in the previous table, it should be mapped against the related QMS internal process covering that subject. Since many organizations adopt an ISO management system (e.g. ISO 9001:2008), a cross-check for validating potential improvements from the design phase could be achieved through re-applying the related mapping document to their own internal process (e.g. using the N/P/L/F – Not/Partially/Largely/Fully achieved ordinal scale from CMMI or

Table 4. Two ISO/IEC 15504 reuse-related processes: suggestions for improvements

ISO/IEC 15504 REU processes	Suggested Improvements
REU.2	**Reuse Asset Management**
BP 01 - Define and document an asset management strategy	• People-related aspects, as necessary skills and experience, are to be addressed in the asset management strategy [Management Tool – Factor 2] • The asset management strategy should consider and differentiate the aspects related to the asset development for reuse respect to those related to asset development with reuse [REBOOT]
BP02 - Establish a classification scheme for assets	• The integration between the asset classification scheme and the Configuration Management rules is to be addressed [Management Tool – Factor 10]
BP 03 - Define criteria for assets	• Possible measurements, to be used as a basis for the definition of criteria for assessment, are to be identified and documented [RiSE MM – Process Factors; RCM] • The determination of asset value should be the basis for the criteria definition [RCM]
BP04 - Establish the asset storage and retrieval mechanisms	• The technology support for storage and retrieval is to be defined [RiSE MM - Technological Factors]
BP 05 - Identify reusable assets.	• The integration between the asset identification and the Configuration Management rules is to be defined [Management Tool – Factor 10]
BP06 - Accept reusable assets	• The technological support for classification and record of assets as well as for their provision to the intended users is to be addressed [Management Tool – Factor 13; RiSE MM -Technological Factors] • Assets integrability is an issues to be addressed [RCM]
BP 07 - Operate asset storage	• The technological support for storage of assets is to be addressed [Management Tool – Factor 13; RiSE MM -Technological Factors]
BP08 - Record use of assets	• -
BP 09 - Notify re-users of asset status	• The notification should rely on established communication channels and an adequate organizational support [Management Tool – Factor 4 and 15]
BP10 - Retire assets	• The technical aspects of the withdrawal from the repository are to be addressed [Management Tool – Factor 13]
REU.3	**Reuse Program Management**
BP 01 - Define organizational reuse strategy	• The top management support is to be explicitly given at organizational reuse strategy definition time [Management Tool – Factor 2] • People-related aspects, as necessary skills and experience, are to be addressed in the reuse strategy [Management Tool – Factor 2] • The organizational reuse strategy should consider and differentiate the aspects related to the asset development for reuse as respect to those related to asset development with reuse [REBOOT] • The reuse organizational reuse strategy should indicate whether and at what extent the product Families approach is adopted [RiSE MM – Business Factors] • The training and education initiatives and activities should be included within the reuse program items [RiSE MM – Business Factors]
BP 02 - Identify domains for potential reuse	• -
BP 03 - Assess reuse capability	• The assessment of the reuse capability of the organization should include cost benefits analysis [Management Tool – Factor 6]
BP 04 - Assess domains for potential reuse	• A measurement scheme should be provided to support of the evaluation of the level of similarities among products in a certain domain [Management Tool MM – Factor 1]
BP 05 - Evaluate reuse proposals	• A measurement scheme should be provided to support of the evaluation of suitability of reusable items [RiSE MM – Process Factors; RCM]
BP 06 - Implement the reuse program	• ---
BP 07 - Collect and manage learning	• Details on the way learning is stored in the repository are to be provided [Management Tool – Factor 13] • The integration between the learning storage and the communication channels for spreading such a knowledge is to be addressed [Management Tool – Factor 4]
BP 08 - Get feedback from reuse	• Communication tool support is to be addressed [Management Tool – Factor 15]
BP 09 - Monitor reuse	• Monitoring is to be included in the reuse planning [Management Tool – Factor 5; RiSE MM – Organizational Factors]

SPICE). In our case, moving from CMMI-DEV, it could be used the Mutafeljia & Stromberg's mapping [36] as a basis. In this paper, our focus was limited to only the design phase. However, a case study with the application of the hybrid-REU processes will be included in a future paper.

The EoI presented in Table 3 as well as the included elements respect the BPs of the REU.2 and REU.3 processes provided in Table 4 are not to be considered exhaustive. These two tables are on the contrary to be considered as a starting point for the application of the LEGO approach in practice.

5 Conclusions and Prospects

Software reuse is the process aimed at defining a set of systematic operating procedures to specify, produce, classify, retrieve, and adapt software artifacts for the purpose of creating software systems from them. Even if there are many existing reuse management models and frameworks, each one represents only one possible view of the inner reality that would be captured and reused: the '*one size doesn't fit all*' motto could be rephrased as '*one model doesn't fit all*'. Thus, at least 2 (or more) models/frameworks should be considered for improving your own processes (whatever they are), in the areas/issues needed.

In order to cope with this need, we recently proposed **LEGO** (**L**iving **E**n**G**ineering pr**O**cess) as an open approach for improving the processes of a business process model (BPM), based upon the comparative analysis of the process architecture and elements of several concurrent models within a certain domain. Since estimation is one of the key processes for determining the success of an organization, we applied LEGO to Reuse, practices with the aim to improving the current ISO/IEC 15504 REU processes by integrating it with other reuse-related maturity models. The final result was the design of a more encompassing hybrid-REU processes that could help organizations to improve their estimates from the beginning of the value chain as well as their construction practices, in order respectively to save time and create more consistent systems.

In the future, we will apply this hybrid-REU processes to real case studies, proposing it as the meta-model to be used for the performing the initial gap analysis against the organizations' BPM related processes as part of an improvement initiative. Another action will be to refine the search of further reuse MMs, trying to catch information also related to ISO 15504 Process Attributes (PAs) and not only Base Practices (BPs), as initially done in this paper (e.g. some technological element supporting better performances).

'*Creativity is allowing yourself to make mistakes.*
Art is knowing which ones to keep,
Scott Adams (1957-)

Acknowledgements. This work has been also supported by the CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico – www.cnpq.br), an entity of the Brazilian government focused on scientific and technological development.

This research is supported in part by the Science Foundation Ireland (SFI) Stokes Lectureship Programme, grant number 07/SK/I1299, the SFI Principal Investigator Programme, grant number 08/IN.1/I2030 (the funding of this project was awarded by Science Foundation Ireland under a co-funding initiative by the Irish Government and European Regional Development Fund), and supported in part by Lero (http://www.lero.ie) grant 10/CE/I1855.

References

[1] Constant, D.: Re: CMMI Representations, which one is the better? Yahoo SPI Mailing List (February 10, 2004), http://goo.gl/5uhAP
[2] Crosby, P.B.: Quality is free. McGraw-Hill (1979) ISBN 0-451-62585-411
[3] CMMI Product Team, CMMI for Development, Version 1.3, CMMI-DEV v1.3, Continuous Representation, CMU/SEI-2010-TR-033. Technical Report, Software Engineering Institute (November 2010)
[4] Curtis, B., Hefley, W., Miller, S.: People Capability Maturity Model (P-CMM) Version 2.0. 2/ed, CMU/SEI-2009-TR-003, Maturity Model, Software Engineering Institute (July 2009), http://goo.gl/2p0M8
[5] SEI, Standard CMMI Appraisal Method for Process Improvement (SCAMPI), version 1.3: Method Definition Document, Software Engineering Institute, Handbook, CMU/SEI-2011-HB-001 (March 2011), http://goo.gl/18IAX
[6] ISO/IEC, IS 15504-2: Information technology – Process assessment – Part 2: Performing an assessment (October 2003)
[7] ISO/IEC, IS 15504-5: Information technology – Process Assessment – Part 5: An exemplar Process Assessment Model (March 2006)
[8] Copeland, L.: The Maturity Maturity Model (M3). Guidelines for Improving the Maturity Process, StickyMinds (September 2003), http://goo.gl/MgUS2
[9] Abran, A., Moore, J.W., Bourque, P., Dupuis, R., Tripp, L.T.: Guide to the Software Engineering Body of Knowledge (SWEBOK) 2004 Version. IEEE (2004), http://goo.gl/OhLDp
[10] ISO/IEC, IS 12207: Information technology – Software Life Cycle processes (2008)
[11] Ibrahim, L., Bradford, B., Cole, D., LaBruyere, L., Leinneweber, H., Piszczech, D., Reed, N., Rymond, M., Smith, D., Virga, M., Wells, C.: The Federal Aviation Administration Integrated Capability Maturity Model, (FAA-iCMM), Version 2.0. An Integrated Capability Maturity Model for Enterprise-wide Improvement, FAA (September 2001)
[12] Burnstein, I., Homyen, A., Grom, R., Carlson, C.R.: A Model to Assess Testing Process Maturity, Crosstalk. The Journal of Defense Software Engineering, 26–30 (November 1998), http://goo.gl/xg8zF
[13] Koomen, T., Pol, M.: Test Process Improvement: a Practical Step-by-Step Guide to Structured Testing. Addison-Wesley (1999) ISBN 0-201-59624-5
[14] OGC, P3M3: Portfolio, Programme & Project Management Maturity Model, Version 1.0, Office of Government Commerce (February 2006), http://goo.gl/tTbq9
[15] PMI, Organizational Project Management Maturity Model (OPM3), Knowledge Foundation, Project Management Institute, 2nd edn. (2008)
[16] CMMI Product Team, CMMI for Service, Version 1.3, CMMI-SVC v1.3, CMU/SEI-2010-TR-034. Technical Report, Software Engineering Institute (November 2010)
[17] CMMI Product Team, CMMI for Acquisition, Version 1.3, CMMI-ACQ v1.3, CMU/SEI-2010-TR-032, Technical Report, Software Engineering Institute (November 2010)

[18] Cardoso Garcia, V., Lucredio, D., Alvaro, A., Santana de Almeida, E.: Towards a Maturity Model for a Reuse Incremental Adoption. In: SBCARS 2007, Brazilian Symposium on Software Components, Architectures and Reuse, http://goo.gl/DVHP9
[19] Buglione, L.: Leveraging people-related maturity issues for achieving Higher Maturity & Capability Levels. In: Proceedings of IWSM/MENSURA 2009, Amsterdam, Netherlands, November 4-6, pp. 35–47 (2009)
[20] Buglione, L.: Maturity Models: modelli esclusivi o integrabili?, Qualità On-Line, Rivista dell'AICQ (Novembre 2007), http://goo.gl/5xvKQ
[21] Jasmine, K.S., Vasanth, R.: A New Capability Maturity Model For Reuse Based Software Development process. IACSIT International Journal of Engineering and Technology 2(1) (February 2010), http://goo.gl/1KE18
[22] Spoelstra, W.: Reusing software assets in agile development organizations - a management tool: a case at a medium sized software development organization. University of Twente, Netherlands. Thesis (2010), http://essay.utwente.nl/59917/
[23] Sindre, G., Conradi, R., Karlsson, E.A.: The REBOOT Approach to Software Reuse, Journal of Systems & Software (JSS). Special Issue on Software Reuse 30(3), 201–212 (1995), http://goo.gl/Sa2Eo
[24] Frakes, W., Terry, C.: Software Reuse: Metrics and Models. ACM Computing Surveys 28(2) (June 1996), http://goo.gl/6mBR4
[25] Reuse Research Center, Software Reuse Fundamentals, Presentation, http://goo.gl/OYWHt
[26] Koltun, P., Hudson, A.: A Reuse Maturity Model. In: WISR4 4th Workshop on Institutionalizing Software Reuse, Center for Innovative Technology, Reston, Virginia, USA (November 1991)
[27] Lloréns, J., Fuentes, J.M., Prieto-Diaz, R., Astudillo, H.: Incremental Software Reuse. In: Morisio, M. (ed.) ICSR 2006. LNCS, vol. 4039, pp. 386–389. Springer, Heidelberg (2006)
[28] Basset, P.G.: The Theory and Practice of Adaptive Reuse. In: Symposium on Software Reusability (SSR 1997). ACM, Boston (1997)
[29] Davis, T.: The reuse capability model: a basis for improving an organization's reuse capability. In: Proceedings of the Second International Workshop on Software Reusability, Herndon, VA (1993)
[30] Griss, M.L.: CMM as a Framework for Adopting Systematic Reuse. Object Magazine, 60–62, 69 (1998), http://goo.gl/k0iXI
[31] Mandava Kranthi, K., Konda, B.M., Thammi Reddy, K., Ravi Kiran, B., Vindhya, A.: A Systematic Mapping Study on Value of Reuse. International Journal of Computer Applications (0975 – 8887) 34(4), 37–44 (2011), http://goo.gl/3oEpm
[32] GUFPI-ISMA, Linee Guida per l'uso Contrattuale dei Function Point, Documento Tecnico 2006-01, Gruppo Utenti Function Point Italia – Italian Software Metrics Association (June 2006)
[33] Buglione, L., Gresse von Wangenheim, C., Hauck, J.C.R., McCaffery, F.: The LEGO Maturity & Capability Model Approach. In: Proceedings of the 5th World Congress on Software Quality, Shanghai, China (October 2011)
[34] Basili, V.R., Caldiera, G., Rombach, H.D.: The Experience Factory. In: Marciniak, J.J. (ed.) Encyclopedia of Software Engineering, vol. 1, pp. 469–476. John Wiley & Sons, Inc. (1994), http://goo.gl/DZIlU
[35] Lim, W.C.: Managing Software Reuse, 1st edn. Prentice-Hall, ISBN 9780135523735
[36] Mutafeljia, B., Stromberg, H.: Process Improvement with CMMI v1.2 and ISO Standards. Auerback Publications (2008), http://goo.gl/BFUqq