Bridging the Gap between Security/Risk Assessment and Quality Evaluation Methods

Luis Olsina[1], Elena Pesotskaya[2], Alexander Dieser[1], and Guillermo Covella[1]
[1]GIDIS_Web, Engineering School at Universidad Nacional de La Pampa (UNLPam), General Pico, Argentina
{olsinal}@ing.unlpam.edu.ar; {alexander.dieser, guillermo.covella}@gmail.com
[2]School of Software Engineering at National Research University Higher School of Economics, Moscow, Russia
{epesotskaya}@hse.ru

Abstract— **An IT security vulnerability can be considered as an inherent weakness in a target system that could be exploited by a threat source. Most vulnerable capabilities/attributes of a system can be identified for instance with security controls in order to evaluate the level of their weaknesses. Thus, understanding the current quality acceptability level achieved for vulnerable attributes can help in turn assessing the risk and planning actions for improvement, i.e. the risk reduction by implementing the risk treatment from the impact standpoint. The underlying hypothesis in our proposal is that each identified attribute associated with the target entity to be controlled should show the highest quality satisfaction level as an elementary indicator. The higher the quality indicator value achieved per each attribute, the lower the vulnerability indicator value and therefore the potential impact. In the present work, we discuss the added value of supporting the IT security and risk assessment areas with measurement and evaluation methods and strategy, which are based on metrics and indicators. Also we illustrate excerpts of an actual case study for measurement and evaluation of a system security characteristic and attributes, and its potential risk assessment.**

Keywords- *IT security vulnerability; risk assessment; measurement; evaluation; metric; indicator.*

Аннотация— **Уязвимости информационной безопасности, вызванные различными угрозами, традиционно являются слабым местом в целевой системе. Многие свойства/атрибуты уязвимостей системы могут быть успешно выявлены, например, с помощью мер контроля безопасности, при помощи которых можно установить также степень подверженности риску. Таким образом, понимание текущего уровня качества атрибутов уязвимости может помочь своевременно выявить риски и предпринять действия для улучшения ситуации, например, снизить риск, изменив его влияние. В статье предложен подход, в основу которого легла гипотеза, что каждый выявленный атрибут, связанный с целевым объектом, находящимся в зоне контроля безопасности, должен отражать наиболее высокий уровень качества как базовый индикатор. Чем выше достигнутое качество индикатора для каждого атрибута, тем ниже будет значение индикатора уязвимости и возможное негативное влияние. В настоящей работе обсуждается эффект от использования методов оценки и измерения уязвимостей в области информационной безопасности и оценке рисков, предлагается стратегия, основанная на использовании метрик и индикаторов. В статье приведены практические кейсы, рассматривающие применение метрик и индикаторов для процессов измерения и оценки уязвимостей и рисков в области информационной безопасности.**

Keywords- *уязвимости информационной безопасности; оценка рисков; метрика; индикатор; система измерения и оценки.*

I. INTRODUCTION

There exist widespread sayings such as "you cannot control what you cannot measure" meaning –as a possible interpretation- that the lack of data e.g. numbers jeopardizes the very basic engineering and management principles of being systematic and disciplined; and "if you do not know where you are, a map will not help to reach the destination" meaning for instance that the lack of data and information for understanding the current situation of an entity vanishes any effort –even having enough resources- to meet the ultimate goal such as improvement. In other words, we cannot improve what we cannot understand, and we cannot appropriately understand without analyzing consistent data and information. So data and information are basic supplies for different processes; while data usually come from facts, measures, formula calculations, etc. –that are often organized as data sets and represented in databases-, information is the meaningful interpretation of data for a given purpose, user viewpoint and context.

In the present work we argue that metrics and indicators are basic, yet key organizational assets for providing suitable data and information for analyzing, recommending, controlling and monitoring. With the aim to systematically carry out measurement and evaluation (M&E) projects and programs, software organizations should establish clearly a set of principles, activities, methods and tools to specify, collect, store, and use trustworthy metrics and indicators and their values. Moreover, in order to make the analysis and decision-making process more robust, it is necessary to ensure that measures and indicators values are repeatable and comparable among the organization's projects. Therefore, it should be mandatory to store not only measurement and evaluation data but also metrics and indicators metadata as for example measurement method, scale, scale type, unit, indicator model, acceptability levels, among others.

In fact, metrics and indicators should be seen as designed and versioned by-product or resources stored in an organizational repository [20]. Particularly, the metric is the sound specification of a measurement process which transforms an entity attribute (i.e. a single quality), the input into a measure (i.e. data), the output; and the elementary indicator is the sound specification of an evaluation process, which has as input a metric's measure and produces an indicator value (i.e. information). However, looking at recognized literature [5, 8, 11, 13, 14, 18] what a metric or indicator means and fits in a given M&E project as well as issues such as *why*, *what*, *who*, *when*, *where* and *how* (W5H for short) to measure and evaluate are very often poorly linked and specified. To make things a bit more complicated, we have observed very often a lack of a sound consensus among M&E terminological bases in different recognized standards and manuscripts or, sometimes, absent terms [19].

Particularly, we emphasize in this work the metric and indicator specification for vulnerable attributes regarding the *Security* characteristic [9] for an information system as target entity. A vulnerability is an inherent weakness in a target system that could be exploited by a threat source. Most vulnerable capabilities/attributes of a system can be identified for instance with security controls either which have not been applied or which, while applied, retain some weakness [17]. Therefore, understanding the current quality acceptability level achieved for vulnerable attributes can help in turn assessing the risk and planning actions for improvement, i.e. the risk reduction by implementing the risk treatment from the impact viewpoint. The underlying hypothesis is that each meaningful attribute associated with the target entity to be controlled should show the highest quality level of satisfaction as an elementary nonfunctional requirement. The higher the quality indicator value achieved per each attribute, the lower the vulnerability indicator value and therefore the potential impact.

Ultimately, the particular contributions of this paper are: i) the awareness of the added value of supporting the IT security/risk assessment area with quality evaluation methods and strategy, which are based on metrics and indicators; ii) a thorough discussion about the specification of metrics and indicators as informational resources for M&E process descriptions, highlighting the importance of recording not only data sets and information but also the associated metadata of information needs, context, attributes, metrics and indicators in order to ensure repeatability and consistently among organization's projects; and iii) the illustration of metrics and indicators from excerpts of an actual IT security and risk evaluation case study. These informational resources are part of an integrated strategy so-called GOCAME (*Goal-Oriented Context-Aware Measurement and Evaluation*) [20, 21], which can be used to understand and improve the quality or capability quality of any organizational entity or asset.

Following this introduction, Section II provides an overview of the GOCAME strategy, focusing on its M&E conceptual framework and process for better understanding the modeling of metrics and indicators. Also an abridged presentation of risk assessment is made in order to consider where M&E are. Section III elaborates on the GOCAME framework and process to precise where the above mentioned W5H issues fit in the M&E process; then, concrete metric and indicator templates for security attributes are fleshed out, following a discussion of our approach contributions. Section IV revises related work and, finally, Section V draws the main conclusions and outlines future work.

II. Background

Measurement and Analysis is for example a basic CMMI process area (at level 2 for the staged capability maturity representation [5]) intended to give support to other process areas by means of measures. Therefore measures and their interpretation for a given information need are considered a key supply to assist and analyze the rest of all other process areas.

Moreover, in order to support consistently in different measurement, evaluation and analysis projects and programs, well-established M&E strategies are needed as well. In [21], two integrated strategies viz. GQM$^+$Strategies [2], and GOCAME were thoroughly analyzed. GOCAME is based on three main principles or capabilities, namely: i) a *conceptual framework* utilizing a robust terminological base; ii) a well-defined *M&E process*; and iii) *evaluation methods and tools*.

GOCAME's first principle is that designing and implementing a M&E project/program requires a sound *M&E conceptual framework*. Often times, organizations conduct start and stop measurement programs because they don't pay enough attention to the way nonfunctional requirements, contextual properties, metrics and indicators should be designed, implemented and analyzed. Any M&E effort requires a M&E framework built on a rich conceptual base, i.e., on an ontological base, which explicitly and formally specifies the main agreed concepts, properties, relationships, and constraints for a given domain. To accomplish this, we developed the C-INCAMI (*Contextual-Information Need, Concept model, Attribute, Metric and Indicator*) framework and its components [20] based on our metrics and indicators ontology [19].

GOCAME's second principle requires a well-established *M&E process* in order to guarantee repeatability in performing activities and consistency of results. A process prescribes a set of phases, activities, inputs and outputs, sequences and parallelisms, roles, check points, and so forth. In [3], a process model for GOCAME was proposed which is also compliant with both the C-INCAMI conceptual base and components.

Finally, GOCAME's third principle is *methods and tools*, which can be instantiated from both the conceptual framework and process. While activities state 'what' to do methods, on the other hand, describe 'how' to perform these activities, which in turn can be automated by tools.

Next, we outline GOCAME's M&E conceptual framework and general process for better understanding the metric and indicator modeling, later on, in Section III.

A. GOCAME Strategy Overview

GOCAME is a multi-purpose strategy that follows a goal-oriented and context-sensitive approach in defining and performing M&E projects. GOCAME is a multi-purpose strategy because can be used to evaluate (i.e. "understand", "improve", etc.) the quality for not only product, system and system-in-use entity categories but also for other ones such as resource and process, by using their instantiated quality models accordingly. Moreover, the evaluation focus can vary, i.e. ranging from "external quality", "capability quality" –or even "non-vulnerability"- to "cost" or "quality/cost" trade-off.

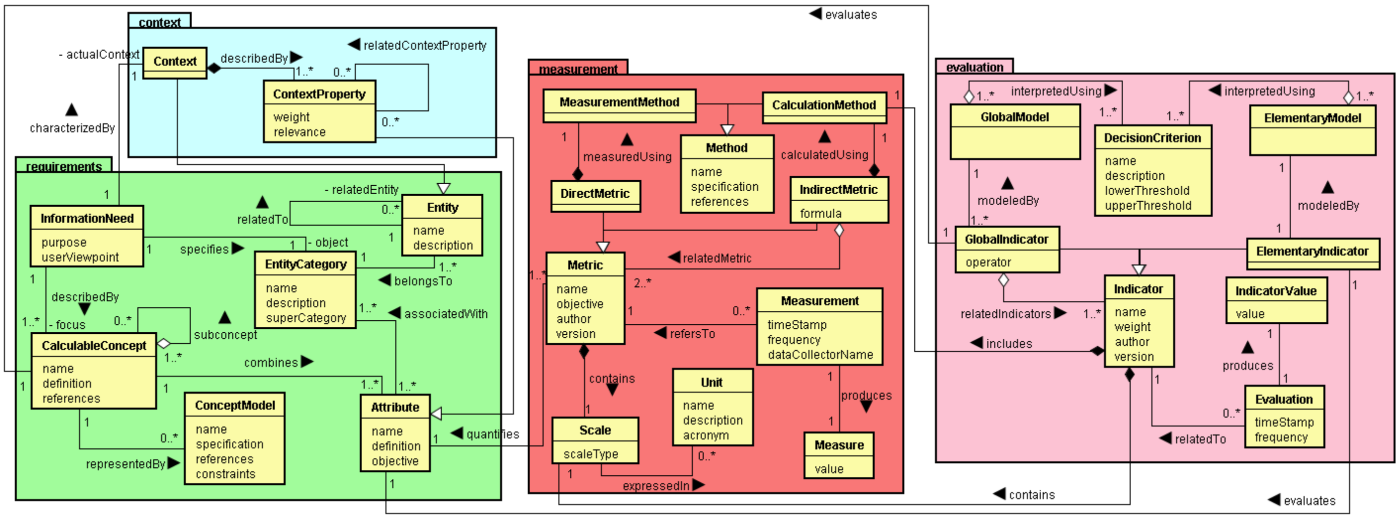

Figure 1. Main concepts and relationships for the C-INCAMI *Requirements*, *Context*, *Measurement* and *Evaluation* components.

Regarding the abovementioned first principle, it has its terminological base defined as an ontology from which emerges the C-INCAMI conceptual framework. The framework is structured in six components, namely: i) *M&E project definition*, ii) *Nonfunctional requirements specification*, iii) *Context specification*, iv) *Measurement design and implementation*, v) *Evaluation design and implementation*, and vi) *Analysis and recommendation specification*. For space reasons, just the components shown in Fig. 1 are presented below, and some key terms are defined as well –for more details see [20].

The *Nonfunctional requirements specification* component (**requirements** package in Fig. 1) allows specifying the *Information Need* of any M&E project. It identifies the *purpose* (e.g. "understand", "predict", "improve", "control", etc.) and the *user viewpoint* (e.g. "developer", "risk manager", "security administrator", etc); in turn, it focuses on a *Calculable Concept* –e.g. instances such as "quality", "security", "reliability"- and specifies the *Entity Category* to evaluate –e.g. a resource, product, system, etc. The leaves of an instantiated model (so-called requirements tree) are *Attributes* associated with an *Entity Category*. From the quoted terms, ***Information Need*** is defined as "*insight necessary to manage objectives, goals, risks, and problems*"; ***Entity Category*** is defined as *"object category that is to be characterized by measuring its attributes or properties"*, and ***Entity*** as *"a concrete object that belongs to an entity category"*. Lastly, ***Attribute*** is "*a measurable physical or abstract property of an entity category*".

Regarding the *Context specification* component (see **context** package in Fig. 1), one key term is ***Context***, which is defined as "*a special kind of entity representing the state of the situation of an entity, which is relevant for a particular information need*". We consider *Context* as a special kind of *Entity* in which related relevant entities are involved. Related entities can be resources –as a network or security infrastructure-, the environment, organization or the project itself, among others. To describe the context, *Attributes* of the relevant entities are used for further quantification, which are called *Context Properties* (see details in [15]).

The *Measurement design and implementation* component allows specifying the metrics that quantify attributes. To design a *Metric*, the *Measurement* and *Calculation Method* and the *Scale* should be defined. Whereas a measurement method is applied to a *Direct Metric,* a calculation method is applied to the *formula* of an *Indirect Metric*. A *Measurement* produces a *Measure* as shown in the *measurement* package in Fig. 1. ***Measurement*** is defined as "*an activity that uses a metric definition in order to produce a measure's value*", while a ***Measure*** is "*the number or category assigned to an attribute of an entity by making a measurement*", and the ***Metric*** is "*the defined measurement or calculation method and the measurement scale*". Hence, for designing a direct metric two aspects should be clearly specified as metadata, namely: its measurement method and scale. The ***Measurement Method*** – synonyms: Counting Rule, Protocol- is "*the particular logical sequence of operations and possible heuristics specified for allowing the realization of a direct metric description by a measurement*"; and the ***Scale*** is "*a set of values with defined properties*". Note that the *scale Type* depends on the nature of the relationship between values of the scale, such as keeping the order and/or distances among categories, in addition to the existence of the zero category (or class) meaning absence of the measured attribute. The scale types mostly used in software engineering are classified into "nominal", "ordinal", "interval", "ratio", and "absolute". It is also important to note that the nominal and ordinal scales do not provide categories that have strict numerical meaning, and for this reason their values are called categorical or qualitative. Conversely, given that the interval, ratio and absolute scale types do provide categories that have numerical meaning, their scale values are called numerical or quantitative. Ultimately, each scale type determines the choice of suitable mathematical operations and statistics techniques that can be used to analyze the data.

The *Evaluation design and implementation* component (**evaluation** package in Fig. 1) includes the concepts and relationships intended to specify the design and implementation of elementary and global evaluations. It is worthy to mention that the selected metrics are useful for a measurement process as long as the selected indicators are useful for an evaluation process in order to interpret the stated information need. *Indicator* is the main term, which allows specifying how to calculate and interpret the attributes and calculable concepts of a nonfunctional requirements tree. There are two types of indicators. First, *Elementary Indicators* that evaluate lower-level requirements, namely, attributes combined in a concept model. Each elementary indicator has an *Elementary Model* that provides a mapping function from the metric's measures (the domain) to the indicator's scale (the range). The new *Scale* is interpreted using agreed *Decision Criteria,* which help to analyze the level of satisfaction reached by each elementary nonfunctional requirement, i.e. by each attribute. Second, *Partial/Global Indicators,* which evaluate mid-level and higher-level requirements, i.e. sub-characteristics and characteristics in a concept model (e.g. a security model). Different aggregation models (*Global Model)* can be used to perform evaluations. The global indicator's value ultimately represents the global degree of satisfaction in meeting the stated information need for a given purpose and user viewpoint. As for the implementation, an *Evaluation* represents the activity involving a single calculation, following a particular indicator specification –either elementary or global-, producing an *Indicator Value.* In our ontology ***Evaluation*** is defined as "*activity that uses an indicator definition in order to produce an indicator's value*", and ***Indicator*** (synonym: Criterion) as "*the defined calculation method and scale in addition to the model and decision criteria in order to provide an estimate or evaluation of a calculable concept with respect to defined information needs*"; lastly ***Decision Criterion*** (synonym: Acceptability Level) is defined as "*thresholds, targets, or patterns used to determine the need for action or further investigation, or to describe the level of confidence in a given result*".

Taking into account the GOCAME's second principle, its general process embraces the following main (core) activities: i) *Define Nonfunctional Requirements*; ii) *Design the Measurement*; iii) *Design the Evaluation*; iv) *Implement the Measurement*; v) *Implement the Evaluation*; and vi) *Analyze and Recommend*. These high-level activities as well as sequences, parallelisms, inputs and outputs are shown in Fig. 2.

The proposed M&E process follows a goal-oriented approach. (Note that a fine-grained specification of this process is in [3]). Once the requirements project has been created, first, the *Define Nonfunctional Requirements* activity has a specific goal, problem or risk as input and a Non-functional Specification document as output (which contain the M&E purpose, user viewpoint, focus, entity, instantiated characteristics and attributes, and context information).

Then, the *Design the Measurement* activity allows identifying the metrics from the Metrics repository to quantify attributes: the output is a Metrics Specification document (each metric specification describes the measurement method, the scale, and other metadata). Note that repositories are represented by <<*datastore*>> stereotype in Fig. 2. Once the measurement was designed, the evaluation design and the measurement implementation activities can be performed –in any order or in parallel. The *Design the Evaluation* activity allows identifying Indicators in order to know the satisfaction level achieved by elementary and global requirements.

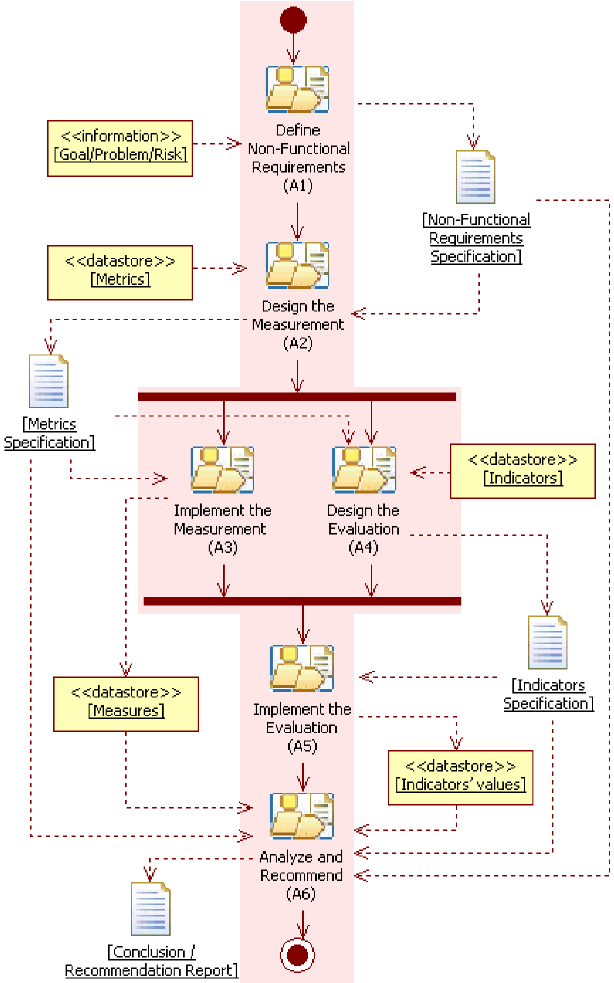
<<information>>
[Goal/Problem/Risk]
Define Non-Functional Requirements (A1)
[Non-Functional Requirements Specification]
<<datastore>>
[Metrics]
Design the Measurement (A2)
[Metrics Specification]
Implement the Measurement (A3)
Design the Evaluation (A4)
<<datastore>>
[Indicators]
<<datastore>>
[Measures]
Implement the Evaluation (A5)
[Indicators Specification]
<<datastore>>
[Indicators' values]
Analyze and Recommend (A6)
[Conclusion / Recommendation Report]

Figure 2. High-level activities for the GOCAME M&E process. Legend A means Activity

The *Implement the Measurement* activity uses the specified metrics to obtain the measures, which are stored in the Measures repository. Next, the *Implement the Evaluation* activity can be carried out. Finally, *Analyze and Recommend* activity has as inputs the values (i.e., data) of measures and indicators, the requirements specification document, and the associated metrics and indicators specifications (i.e., metadata) in order to produce a Conclusion/Recommendation report.

Since the M&E process includes activities such as specify the requirements tree, identify metrics, analyze and recommend, and so on, it is necessary to have a methodology that integrates all these aspects and tools that automate them; that is to say, a set of well-defined and cooperative methods, models, techniques and tools that, applied consistently to the process activities produces the different outcomes.

To this aim the WebQEM (*Web Quality Evaluation Method*) methodology and its associated tool (named C-INCAMI_Tool [20]) were instantiated from the conceptual framework and process, and used in different academic and industrial case studies.

It is important to remark that some GOCAME methods for evaluation are based on multi-criteria (attribute) decision analysis, which can also be used for risk assessment [12].

B. Risk Assessment Issues regarding M&E

There are abundant standards and research (e.g. [1, 5, 10, 12, 17], to quote just a few) in areas of risk management, risk assessment techniques and processes as well as risk vocabularies. However, an ontology for risk management as we did for metrics (measurement) and indicators (evaluation) is to the best of our knowledge missing yet, so we consider its development as future work. In this paper, without entering in the specific discussion of the risk terminological base, we consider to use some terms defined in the previous sub-section as entity, entity category, attribute, contextual property, etc.

Categories of entities as for example development/ maintenance/service software projects, products and systems or some of their components, among others, involve risks at different development or operative stages which should be identified, prevented, controlled, treated and monitored through a well-defined and systematic risk management approach. A risk can be defined as an undesirable consequence of an event on a target entity, which can represent an organizational asset – where an asset is an entity with (added) value for an organization. The potential losses affecting the asset are also called impact of the risk. In addition, the term vulnerability is commonly used –for instance in security-, which briefly means a weakness of an entity that can be exploited by a threat source.

Software Risk Management (SRM) suggests actions e.g. to prevent risks or to reduce its impact on the target entity instead of dealing with its further consequences. Thus, we can identify the most relevant attributes associated to an entity which can be more vulnerable (weak) from triggered external/internal events. Then, by understanding the current attributes' strengths and weaknesses –i.e. by using an evaluation-driven approach as GOCAME-, actions for change can be recommended and planned for further treatment implementation.

In general terms, SRM includes a set of policies, processes, methods, techniques and tools to analyze risks, understand weaknesses, prepare preventive/perfective actions, and control the risks on the target entity. Particularly, for *risk assessment* three general activities are proposed in [12] namely: i) *Risk Identification*; ii) *Risk Analysis*; and iii) *Risk Evaluation*. In addition, *Establishing the context*, *Risk treatment*, *Risk monitoring and review*, and *Communication* are common processes for a well-established SRM strategy as well. Basically, the *Risk Identification* activity aims at gathering information about all risks which can affect the information system or resource (i.e. the target entity), such as risk category, possible causes and outcomes, etc. In [10] it is defined as "the process of finding, recognizing and describing risks". Also a note indicates "risk identification involves the identification of risk sources, events, their causes and their potential consequences". In the *Risk analysis* activity, the identified risks are prioritized according to the probability of occurrence and loss/undesirable consequences associated to the entity attributes to set how many risks will be treated. It is defined as "the process to comprehend the nature of risk and to determine the level of risk". Also a note indicates "risk analysis provides the basis for risk evaluation and decisions about risk treatment" [10]. *Risk evaluation* activity is defined as "the process of comparing the results of risk analysis with risk criteria to determine whether the risk and/or its magnitude is acceptable or tolerable". Risk evaluation assists in the decision about risk treatment, which is defined as "the process to modify risk" [10]. Usually risk treatment can involve: i) avoiding the risk by deciding not to start or continue with the activity that gives rise to the risk; ii) taking or increasing risk in order to pursue an opportunity; iii) removing the risk source; iv) changing the likelihood (probability); v) changing the consequences; vi) sharing the risk with another party or parties; and vii) retaining the risk by informed decision.

In this work, we propose, for brevity reasons, concentrating just on the v) above item –particularly, in the vulnerabilities- for designing the measurement, evaluation and ulterior improvement plan. The plan should describe actions to reduce the vulnerability/impact on the target entity. Particularly, regarding our proposal, system attributes, metrics and indicators should be selected to manage the risk status, showing whether the risk is reduced. The interpretation of evaluations is made by the used indicators and their acceptability levels met, which in turn make use of metrics that quantifies the attributes associated to the entity as per plan.

Hence, target entities can be measured and evaluated by means of their associated attributes and calculable concepts (characteristics). The underlying hypothesis is that each meaningful attribute to be controlled (related for example to the *Security* characteristic) should show the highest quality level of satisfaction as an elementary nonfunctional requirement. The higher the quality indicator value achieved per each attribute, the lower the vulnerability indicator value. In percentage terms:

Vulnerability Indicator value (for Attribute A_i) = 100 – Quality Indicator value (for A_i), (1)

where in the percentage scale there are acceptability levels for the elementary quality indicator, representing 100% a totally satisfied (achieved) requirement, and 0% totally unsatisfied –so implies that an urgent change action must be planned. So per each relevant attribute A_i, we can calculate the risk value (magnitude) before and after improvement changes were performed using the often quoted formula:

Risk value for A_i = Probability of Event occurrence for A_i * Vulnerability Indicator value for A_i (2)

Then calculate the risk reduction per each vulnerable attribute. Or even the risk reduction calculated as an aggregated indicator value e.g. for the *Security* characteristic.

Ultimately, without the well-established support of metrics and indicators and their values SRM is more craftwork than science! The proposed approach of looking at vulnerabilities as attributes of target entities and then using metrics and indicators for their measurement and evaluation and further analysis is illustrated in sub-section III. B.

III. SECURITY METRIC AND INDICATOR SPECIFICATIONS

One of the stated contributions in the Introduction Section is that metrics and indicators are basic, yet key organizational assets for providing germane data and information for analyzing, recommending, controlling and ultimately making decisions.

From the specification standpoint, metrics and indicators can be considered as designed and versioned by-products stored in organizational repositories, which are used by measurement and evaluation processes. Nevertheless, regarding the state-of-the-art literature, what metrics and indicators mean and where fit in a given M&E process in addition to issues such as *why*, *what*, *who*, *when*, *where* and *how* to measure and evaluate have been often poorly related and specified, as we discuss later in Section IV.

A. The W5H rule: Why, What, Who, When, Where, How?

Nelson [16] asserts that a "discussion of the *why*, *who*, *what*, *where*, and *when* of security metrics brings clarity and further understanding because it establishes a framework for implementing a framework to implement security metrics in your organization" (cf. p.14).

We want to reinforce this idea and try to make it sounder. GOCAME's three principles outlined in Section II.A –viz. the *M&E conceptual framework*, *process* and *methods*- will help us illustrate the rationale for the W5H mnemonic rule. Particularly, in the following summary, we rely on the general process depicted in Fig. 2, which in turn is compliant with the terminological framework shown in Fig. 1.

Why an organization should tackle the M&E endeavor might be materialized in the *M&E project definition* and instantiation –and obviously, supported by a broader quality assurance policy. Basically, there is a problem or issue (see the *goal/problem/risk* input in Fig 2) that requires a solution driven by analysis and decision making. For instance, the organization needs to reduce some particular entity vulnerabilities; however, as commented above, it cannot improve what cannot understand, and cannot appropriately understand without analyzing consistently data and information. The why aspect therefore embraces the concrete *information need* and *purpose* for M&E such as understand, improve, and control some relevant objective, regarding a specific *user viewpoint*.

What is to be measured and evaluated? This embraces the concrete target *entity* –and related entities including *context* that *belongs to* an *entity category*. In addition, a given information need is described by a *focus* (e.g. the security *calculable concept*) to which *attributes* are combined. Moreover, entities cannot be measured and evaluated directly but only by means of their associated attributes and *context properties*. Ultimately, the *non-functional requirements specification* artifact (see Fig. 2) documents to a great extent the why and the what.

How basically deals with the metric and indicator specifications. *Metrics* and *indicators* are organizational assets stored in repositories (as depicted in Fig. 2), which are selected respectively by the A2 and A4 activities at design time, and then implemented by the A3 and A5 activities accordingly. As we show in sub-section III.B, metric and indicator specifications should be considered metadata which must be kept linked –for consistency reasons- to measure and indicator values produced for the A3 and A5 activities. Also metadata and datasets are consumed by the A6 activity as well.

Who is responsible for the different stages of a M&E project? Certainly, there are different levels of responsibilities and roles –e.g. in [4], 13 roles using the GOCAME strategy for a given M&E project are defined. In the C-INCAMI *M&E project definition* component (not shown in Fig. 1) related project concepts allow recording the *responsible* information. In addition, *author* name is a common field for both *metric* and *indicator* specifications which represents their creator as a piece of design. Besides, the *data collector* name –see *measurement* term in Fig 1- allows recording the responsible of data gathering for the A3 activity.

When is recorded for each M&E project and also per each enacted measurement and evaluation. Basic questions supported are, among others: When do you collect metrics? How often do you collect them? When do you perform evaluations and analysis? For example, the *time stamp* and *frequency* fields in the *measurement* and *evaluation* terms allow recording them accordingly when A3 and A5 are executed.

Where is the M&E project running? Where is the entity under evaluation placed? In which context is the target entity measured and evaluated? Where is data collection activity for metrics performed? Some of these raised issues can be taking into account by the C-INCAMI *M&E project definition* component including the recorded *context* and its associated *context properties* and *values*.

In the following sub-section, using the W5H rule we illustrate some *Security* attributes, metrics and indicator for a system, emphasizing mainly the **how** aspect, for space reasons.

B. Security Characteristic for a System: Proof of Concept

In the present sub-section, excerpts of an actual case study we are carrying out are used as proof of concept. One aspect of the mnemonic rule is the issue of **what** is to be measured and evaluated?

In this study the concrete target entity is the "SIU Guarani register system", a student management system widespread used in Argentinean universities. It is an information system –from the entity category standpoint- commonly used by students, professors and faculty members in many schools.

Therefore, **why** should it be evaluated? Because a concrete information need was raised by the IT responsible of the Engineering School at UNLPam, which is related to security risks due to different potential threats, as for example, students changing the bad marks of subjects due to system vulnerabilities. Note that if this threat would materialize, the impact for the institution discredit will be high.

So the purpose of the information need is firstly to understand the current external quality satisfaction level achieved, particularly for the non-vulnerabilities regarding the security feature, from the security administrator user viewpoint. Once the current security satisfaction level is understood, secondly the purpose is to improve the system in those weakly performed indicators. That is to say, the ultimate purpose is to reduce the security risks.

Fig. 2 shows as output of the A1 activity the non-functional requirements specification artifact, which mainly documents the why and what aspects. Specifically, Table I represents the requirements tree instantiated for the *Security* characteristic and its sub-characteristics such as *Confidentiality* (coded 1.1), *Integrity* (1.2) and *Authenticity* (1.3), which are the ones prescribed in the ISO 25010 external quality model [9]. Note that other characteristics and sub-characteristics are being used in the case study, e.g. *Availability* but are not illustrated here for space reasons.

In Table II each sub-characteristic is defined. For instance, *Confidentiality* represents "*the degree to which a product or system ensures that data are accessible only to those authorized to have access"*.

Additionally, we have identified for 1.1 the *Access Schema Protectability* (1.1.1) sub-characteristic which is defined as "*the degree to which the system ensures the confidentiality of data by providing access protection capabilities"*. In turn, three measurable attributes were specified for 1.1.1 as shown in Table I –note that attributes are highlighted in italic.

TABLE I. REQUIREMENTS TREE SPECIFICATION FOR 'SECURITY'

1. Security
1.1. Confidentiality
1.1.1. Access Schema Protectability
1.1.1.1. Authentication Schema Bypass
1.1.1.2. Password Aging Policy
1.1.1.3. String Password Robustness
1.2. Integrity
1.2.1. Cross-Site Scripting Immunity
1.2.1.1. Reflected Cross-Site Scripting Immunity
1.2.1.2. Stored Cross-Site Scripting Immunity
1.2.1.3. DOM-based Cross-Site Scripting Immunity
1.2.1.4. Cross-site request forgery Immunity
1.3. Authenticity
1.3.1. Session Impersonation Protectability
1.3.1.1. Session Data Disclosure Protectability
1.3.1.2. Session ID Disclosure Protectability
1.3.1.3. Session Non-Replay Protectability

TABLE II. DEFINITION OF THE 'SECURITY' CHARACTERISTIC AND ITS USED SUB-CHARACTERISTICS

Calculable Concept	Definition
Security (coded 1 in Table I)	Degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization [9].
Confidentiality (1.1)	Degree to which a product or system ensures that data are accessible only to those authorized to have access [9].
Access Schema Protectability (1.1.1)	Degree to which the system ensures the confidentiality of data by providing access protection capabilities.
Integrity (1.2)	Degree to which a system, product or component prevents unauthorized access to, or modification of, computer programs or data. ISO Note [9]: Immunity (the degree to which a product or system is resistant to attack) is covered by integrity.
Cross-Site Scripting Immunity (1.2.1)	Degree to which the system ensures the integrity of data by providing cross-site scripting immunity capabilities.
Authenticity (1.3)	Degree to which the identity of a subject or resource can be proved to be the one claimed [9].
Session Impersonation Protectability (1.3.1)	Degree to which the system ensures session impersonation protection by providing secure session handling protocols.

TABLE III. SPECIFICATION OF ALL METRICS INVOLVED IN QUANTIFYING THE '*AUTHENTICATION SCHEMA BYPASS*' ATTRIBUTE

Attribute:
Name: *Authentication Schema Bypass* **Coded:** 1.1.1.1 in Table I
Definition: Due to negligence, ignorance or understatement of security threats often result in authentication schemes that can be bypassed by simply skipping the login page and directly calling an internal page that is supposed to be accessed only after authentication has been performed.
Objective: To find out the degree to which bypassing the authentication schema is avoided.
Indirect Metric:
Name: *Ratio of Protected Pages Accessed via Forced Browsing* (%PPA)
Objective: To determine the ratio between the number of successful attempts accessing protected pages by forced browsing and the total number of attempts performed.
Author: Covella G. and Dieser A.
Version: 1.0
Reference: OWASP Testing Guide 2008 V3.0
Calculation Method:
Formula Specification: %PPA = (#PF / #TPP) * 100
Numerical Scale:
Representation: Continuous
Value Type: Real
Scale Type: Proportion
Unit:
Name: Percentage
Acronym: %
Related Metrics:
1) *Number of successful attempts to access protected pages by forced browsing* (#PF); and 2) *Total number of attempts to access protected pages by forced browsing* (#TPP)

Attribute: *Amount of successful attempts to access protected pages*
Direct Metric:
Name: Number of successful attempts to access protected pages by forced browsing (#PF)
Objective: The number of successful attempts bypassing the authentication schema for the protected page population using the forced browsing technique
Author: Covella G. and Dieser A.
Version: 1.0
Measurement Method:
Name: Direct page request
Specification: Using an unauthenticated browser session, attempt to directly access a previously selected protected page URL through the address bar in a browser. Add one per each successful access which bypasses the authentication.
Type: Objective
Numerical Scale:
Representation: Discrete
Value Type: Integer
Scale Type: Absolute
Unit:
Name: Successful attempts
Acronym: Sa

Attribute: *Amount of attempts to access protected pages*
Direct Metric:
Name: Total number of attempts to access protected pages (#TPP)
Objective: The total number of protected pages (i.e. the given population) to be attempted for access by a given technique
Author: Covella G. and Dieser A.
Version: 1.0
Measurement Method:
Specification: As precondition, log into the website with a valid user ID and password. Browse the site looking for the URL population of protected pages, which are those that must be accessed only after a successful login. Add one per each protected page URL selected.
Type: Objective
Numerical Scale:
Representation: Discrete
Value Type: Integer
Scale Type: Absolute
Unit:
Name: Protected pages
Acronym: Pp

For example, the objective of the 1.1.1.1 attribute is to find out the degree to which bypassing the authentication schema is avoided. While most applications require authentication for gaining access to private information or to execute tasks, not every authentication method is able to provide adequate security. In Table III this attribute is also defined and its indirect metric –and related direct metrics- is thoroughly represented in the metric template.

As aforementioned for the W5H rule, the **how** issue deals basically with the metric and indicator specifications. Once the nonfunctional requirements were specified, the next A2 activity consists in selecting the meaningful metrics from the Metrics repository (see Fig. 2) to quantify attributes. One metric should be assigned per each attribute of the requirements tree respectively. For example, the indirect metric named *Ratio of Protected Pages Accessed via Forced Browsing* was selected for quantifying the *Authentication Schema Bypass* (1.1.1.1) attribute. The reader can observe in the templates of Table III this indirect metric is composed of two related direct metrics, which are also fully specified.

While an indirect metric has a calculation method for its formula specification, a direct metric has a measurement method. For instance, the measurement method for #PF direct metric is objective, i.e. it does not depend of human judgment when the measurement is performed. The measurement method represents the counting rule and its specification for #PF indicates "using an unauthenticated browser session, attempt to directly access a previously selected protected page URL through the address bar in a browser. Add one per each successful access which bypasses the authentication". In addition, its measurement method can be automated by a software tool, so this field can be added to the metric template as well.

Ultimately, the metric representation as informational resource for the A2 and A4 activities embraces metadata such as scale, scale type, value type, measurement/calculation method specification, tool, version, author, among others. These metric metadata allow therefore repeatability among M&E projects and consistency in the ulterior analysis of data sets. Once all metrics were selected for quantifying the 9 attributes of Table I, next the A4 activity should be performed, which deals with designing the evaluation.

While an elementary indicator evaluates the satisfaction level reached for an elementary requirement, i.e., an attribute of the requirements tree, a partial/global indicator evaluates the satisfaction level achieved for partial (sub-characteristic) and global (characteristic) requirements. As commented in Section II.A, indicator is the main concept for evaluation, which can be elementary or partial/global ones.

In Table IV the elementary indicator named *Performance Level of the Authentication Schema Bypass* is specified. This elementary indicator will determine the quality satisfaction level reached by the 1.1.1.1 attribute considering the measured value of its indirect metric. Conversely to metrics, indicators have decision criteria for data interpretation, which ultimately means information in the context. In Table IV, three acceptability levels useful for the interpretation of indicator values in the percentage scale are employed after an agreement with evaluation stakeholders. A value between zero and sixty ([0-60]) represents an unsatisfactory level and means that *"change actions must be taken with high priority"*; a value between sixty and ninety ([60-90]) represents a marginal level that means that *"improvement actions should be taken"*; while a value between ninety and a hundred ([90-100]) corresponds to the satisfactory acceptability level.

With regard to the **how** for a global indicator –which evaluates characteristics and sub-characteristics of a requirements tree-, it has similar metadata as shown for an elementary indicator. But instead of an elementary model it has a global or aggregation model.

An example of global model is LSP (*Logic Scoring of Preference*), which was used e.g. in [20, 21]. LSP is a weighted multi-criteria aggregation model, which has operators for modeling simultaneity (C –conjunctive- operators) and replaceability (D –disjunctive- operators) relationships among attributes, sub-characteristics and characteristics of a requirements tree. For instance, the C-+ weak conjunction operator lets modeling the simultaneity criterion among the 1.1, 1.2 and 1.3 sub-characteristics, yielding zero if one input were zero. Next, it is the specification of the LSP aggregation model:

$$P/GI\ (r) = (W_1 * I_1^{\,r} + W_2 * I_2^{\,r} + \ldots + W_m * I_m^{\,r})^{1/r}\ , \qquad (3)$$

where *P/GI* represents the partial/global indicator to be calculated, and I_i stands for elementary indicator value and the following holds $0 <= I_i <= 100$ in a percentage scale; W_i represents the weights, where: $W_1 + W_2 + \ldots + W_m = 1$, and $W_i > 0$ for i = 1 to m; and, *r* is a parameter selected to achieve the desired logical simultaneity or replaceability relationship.

TABLE IV. ELEMENTARY INDICATOR TEMPLATE USED FOR INTERPRETING THE '*AUTHENTICATION SCHEMA BYPASS*' ATTRIBUTE

Attribute: *Authentication Schema Bypass* **Coded:** 1.1.1.1 in Table I
Elemental Indicator:
Name: Performance Level of the Authentication Schema Bypass (P_ASB)
Author: Covella G. and Dieser A.
Version: 1.0
Elementary Model:
Function Name: P_ASB function
Specification: the mapping is: P_ASB = 100 iff $\%PPA < \%PPA_{MIN}$; P_ASB = 80 iff $\%PPA_{MIN} <= \%PPA < \%PPA_{MAX}$; P_ASB = 0 iff $\%PPA >= \%PPA_{MAX}$ where %PPA is the indirect metric specified in Table III.
Decision Criterion:
[Acceptability Levels]
Name 1: Unsatisfactory
Description: indicates change actions must be taken with high priority
Range: if 0 ≤ P_ASB ≤ 60
Name 2: Marginal
Description: indicates a need for improvement actions
Range: if 60 < P_ASB ≤ 90
Name 3: Satisfactory
Description: indicates no need for current actions
Range: if 90 < P_ASB ≤ 100
Numerical Scale:
Representation: Continuous
Value Type: Real
Scale Type: Proportion
Unit:
Name: Percentage
Acronym: %

Lastly, as result of the whole design and selection process –activities A1, A2 and A4 in Fig. 2-, the following documents are yielded: the non-functional requirements specification, the metrics specification and the indicators specification.

Aspects of **when** and **where** are related to great extent to the *Implement the Measurement and Evaluation* activities, as commented in sub-section III.A. Particularly, for each executed M&E project, the A3 and A5 activities produce measure and indicator values accordingly at given moments in time and frequencies.

C. Added Value of Metrics/Indicators for Bridging the Gap

We have illustrated above the specification of a security metric and a quality elementary indicator both regarded as informational resources for M&E process descriptions. Therefore, it is worthy to remark again that metric and indicator specifications should be considered metadata which must be kept linked –for reasons of analysis comparability and consistency- to measure and indicator values (datasets) produced by the A3 and A5 activities.

Let's suppose for example that the same *Authentication Schema Bypass* (1.1.1.1) attribute can be quantified by two metrics (recall in Fig. 1 that an attribute can be quantified for many metrics, but just one must be selected for each specific M&E project from the Metric repository in the A2 activity). So one metric (M1) in the repository is that specified in Table III, and the other metric (M2) is one which has different measurement method and scale type; e.g. M2 considers the predictability of the session identifiers (IDs) as method, and a categorical scale, particularly, an ordinal scale type with values ranging from 1 to 3, where 3 represents the higher difficulty to predict the ID session, and 1 the lower. After many M&E projects using the same nonfunctional requirements –i.e. the security (sub-)characteristics and attributes- are executed, all data and datasets from measurement are recorded in the Measure repository. In some projects were used M1 and in others M2 for quantifying the 1.1.1.1 attribute.

Therefore, if metadata of recorded data were not linked appropriately, e.g. to the measured value 3 which can come from both metrics in different projects, the A6 activity will produce inconsistent analysis if takes as inputs all these related projects. This inconsistency is due to the 3 value, depending on the used metric, has different scale properties recalling that each scale type determines the choice of suitable mathematical operations and statistics techniques that can be used to analyze data and datasets. In summary, even if the attribute is the same, both metric measures are not comparable.

On the other hand, regarding the elementary indicator shown in Table IV, its specification is in terms of quality satisfaction levels –since the *Security* characteristic in Table I is represented by the ISO quality model-, so each vulnerability indicator value can be obtained as per Equation 1. Recall that the underlying hypothesis is that each security attribute to be controlled for the target entity should show the highest quality level of satisfaction as an elementary nonfunctional requirement. But as the reader can surmise, the elementary indicator template in Table IV could also represent the vulnerability level almost straightforwardly, under the premise that the higher the quality indicator value achieved per each attribute, the lower will be the vulnerability indicator value. Hence, the risk value per each vulnerable attribute can be calculated using Equation 2.

Lastly, the aggregation model in Equation 3 can be used for calculating the current state of the security global risk based on risk elementary indicator values. Also the risk reduction can be calculated after improvement actions (risk treatment) and re-evaluation were performed. These issues will be thoroughly illustrated in a follow-up paper.

IV. Related Work

Considering the state-of-the-art research literature, what metrics and indicators mean and where they properly fit in with regard to specific M&E processes and strategy have often been understated or neglected. Furthermore, there are abundant research and standards in areas such as measurement and analysis [2, 5, 8, 13, 14], IT security and risk assessment [1, 11, 12, 17, 18], but issues such as *why*, *what*, *who*, *when*, *where* and *how* to measure and evaluate have also often been poorly intertwined and specified.

For instance, as quoted in sub-section III.A, Nelson states that a "discussion of the *why*, *who*, *what*, *where*, and *when* of security metrics brings clarity and further understanding because it establishes a framework for implementing a framework to implement security metrics in your organization". Nevertheless, in our opinion Nelson fails in discussing the W5H mnemonic rule with more robust conceptual grounds as we did based on GOCAME first and second principles introduced in sub-sections II.A and B. Moreover, the *how* issue –which precisely deals with the key aspect of metric and indicator specifications- is also left aside, when the author remarks "*How* is left as an exercise for the reader" (cf. p.14).

On the other hand, we have developed an integrated M&E strategy so-called GOCAME, which is made up of three capabilities, i.e. the conceptual framework, the process, and the methodology, as overviewed in Section II. The metric and indicator ontology used by the C-INCAMI conceptual framework has similarities to the one presented in [6] and then slightly refined in [7]. However in [19] we have modeled some terms (e.g., elementary indicator, global indicator, etc.) and some relationships (e.g., measurement and measure, metric and indicator, among others) that differ semantically with those proposed in [7]. In addition, we have enlarged the metric and indicator ontology with context terms and relationships [15] while in [7] these are missing. Moreover, GOCAME exhibits a terminological correspondence between the C-INCAMI conceptual framework and the process specifications; for example, the activity diagram of Fig. 2 shows many of the same terms defined in the ontology and depicted in Fig. 1.

Lastly, in order to support repeatability and consistency of results among different measurement, evaluation and analysis projects and programs, well-established M&E strategies are needed as well. In [21] two integrated strategies –which can also be used for risk measurement and evaluation- viz. GQM$^+$Strategies [2] and GOCAME were assessed and analyzed thoroughly. The study drew GQM$^+$Strategies performs lower than GOCAME regarding the suitability of the conceptual base and framework.

Ultimately, the sound and complete specification of metrics and indicators as shown in the previous templates (i.e. in tables III and IV) outperforms the examined ones in the related work.

V. Conclusion and Future Work

For these concluding remarks, we would like to point out main aspects of the three contributions listed in Section I. The first stated contribution that says "the awareness of the added value of supporting the IT security/risk assessment area with quality M&E methods and strategy, which are based on metrics and indicators" was the main drive of the paper. We have argued our approach helps bridging the gap between the IT security/risk assess assessment area and existing quality M&E methods. The entrance gate –as described in sub-section II.B- is based on identifying vulnerable attributes of a target entity, which can be quantified by metrics and interpreted by indicators. Hence, by using an evaluation-driven strategy as GOCAME, we can apply for quality and risk assessment its multi-criteria (attribute) decision analysis methods.

To the second contribution that says "a thorough discussion about the specification of metrics and indicators as informational resources for measurement and evaluation process descriptions…" we have argued that metrics and indicators are basic, yet key organizational assets for providing suitable data and information for analyzing, recommending, controlling and ultimately decision-making processes. Also we have remarked the metric is the sound specification of a measurement process that transforms an entity attribute –the input- into a measure –the output, i.e. data-; and the elementary indicator is the sound specification of an evaluation process, which has as input a metric's measure and produces as output an indicator value –i.e. contextual information. Besides, we have highlighted the importance of recording not only data sets and information but also the associated metadata throughout the paper, giving details in Section III and an example of a potential wrong analysis in sub-section III.C.

Finally, the last stated contribution "the illustration of metrics and indicators from excerpts of an actual IT security and risk evaluation case study" has been made mainly in sub-section III.B. The purpose of the information need is firstly to understand the current quality satisfaction level achieved to the *Security* characteristic for the SIU entity, from the security administrator user viewpoint. Once its current state is understood, the following purpose is to improve the SIU system in those weakly performed indicators; that is, to reduce its security risks. The whole results of this case study, for space reasons, will be documented in a separate paper.

Regarding future work, an ontology for risk assessment as we did for metrics (measurement) and indicators (evaluation) is to the best of our knowledge missing yet, so we are considering its development in the near future. Currently, there exist vocabularies such as in [10], but we are aware that an ontology supports richer conceptual framework modeling than a glossary of terms; hence, this can benefit the instantiation of SRM strategies, processes and methods as well.

Acknowledgment

This work and line of research are supported by the 09/F047 and PAE-PICT 2188 projects at UNLPam, from the Science and Technology Agency, Argentina.

References

[1] Alberts C., Dorofe A. OCTAVE – Method Implementation Guide Version 2.0, Carnegie Mellon, SEI, USA, 2001.

[2] Basili V., Lindvall M., Regardie M., Seaman C., Heidrich J., Jurgen M., Rombach D., Trendowicz A. Linking Software Development and Business Strategy through Measurement, IEEE Computer, (43):4, pp. 57–65, 2010.

[3] Becker P., Molina H., Olsina L. Measurement and Evaluation as quality driver. In: ISI Journal (Ingénierie des Systèmes d'Information), Special Issue "Quality of Information Systems", Lavoisier, Paris, France, (15): 6, pp. 33-62. 2010.

[4] Becker P., Lew P., Olsina, L. Specifying Process Views for a Measurement, Evaluation, and Improvement Strategy. In: Advances in Software Engineering Journal, Acad. Editor: Osamu Mizuno, Hindawi Publishing Co, V. 2012, 27 pg., DOI:10.1155/2012/949746. 2012.

[5] CMMI Product Team. CMMI for Development, Ver.1.3. CMU/SEI-2010-TR-033, USA, 2010.

[6] García F.; Ruiz F.; Bertoa M.; Calero C.; Genero M.; Olsina L.; Martín M.; Quer C.; Condori N.; Abrahao S.; Vallecillo A., Piattini M. An ontology for software measurement, Technical Report UCLM DIAB-04-02-2, Computer Science Department, University of Castilla-La Mancha, Spain, (In Spanish), 2004.

[7] Garcia F.; Bertoa M.; Calero C.; Vallecillo A.; Ruiz F.; Piattini M.; Genero M. Towards a consistent terminology for software measurement. Information and Software Technology (48):8, pp. 631-644, 2005.

[8] Goethert W., Fisher M. Deriving Enterprise-Based Measures Using the Balanced Scorecard and Goal-Driven Measurement Techniques, Software Engineering Measurement and Analysis Initiative, CMU/SEI-2003-TN-024, Available online, 2003.

[9] ISO/IEC 25010. Systems and software engineering – Systems and software product Quality Requirements and Evaluation (SQuaRE) – System and software quality models, 2011.

[10] ISO/IEC Guide 73. Risk management – Vocabulary – Guidelines for use in standards. 2009.

[11] ISO/IEC 27004. Information technology — Security techniques — Information security management — Measurement, 2009.

[12] ISO/IEC 31010. Risk management – Risk assessment techniques, 2009.

[13] ISO/IEC 15939. International Standard, Information technology - Software Engineering: Software Measurement Process, Geneva, Switzerland, 2002.

[14] Kitchenham B., Hughes R., Linkman S. Modeling Software Measurement Data. IEEE Transactions on Software Engineering. (27):9, pp. 788-804, 2001.

[15] Molina H.; Rossi G., Olsina L. Context-Based Recommendation Approach for Measurement and Evaluation Projects, In: Journal of Software Engineering and Applications (JSEA), Irvine, USA, (3): 12, pp. 1089-1106, 2010.

[16] Nelson C. Security Metrics: An Overview, In: ISSA Journal, pp. 12-18, August 2010.

[17] NIST SP 800-30. Guide for Conducting Risk Assessments. Available at http://csrc.nist.gov/publications/PubsSPs.html, Set. 2011, accessed in March, 2012.

[18] NIST SP 800-55. Performance Measurement Guide for Information Security. Available at http://csrc.nist.gov/publications/PubsSPs.html, July 2008, accessed in March, 2012.

[19] Olsina L., Martín M. Ontology for Software Metrics and Indicators. In: Journal of Web Engineering, Rinton Press, USA, (2): 4, pp. 262-281, 2004.

[20] Olsina L., Papa F., Molina H. How to Measure and Evaluate Web Applications in a Consistent Way. Chapter 13 in Springer book: *Web Engineering: Modeling and Implementing Web Applications*, Rossi G., Pastor O., Schwabe D. and Olsina L. (Eds), pp. 385-420, 2008.

[21] Olsina L., Papa F., Becker P. Assessing Integrated Measurement and Evaluation Strategies: A Case Study, In: IEEE Xplore, ISSN 978-1-4673-0844-1/11, 7th Central Eastern European Software Engineering Conference (CEE-SECR 2011), Moscow, Russia, pp. 1-10, 2011.