

Assessing Web Applications Consistently: A Context Information Approach

Hernan Molina and Luis Olsina

GIDIS Web, Engineering School at Universidad Nacional de La Pampa, Argentina
[hmolina, olsinal]@ing.unlpam.edu.ar

Abstract

In order to assess Web applications in a more consistent way we have to deal not only with non-functional requirement specification, measurement and evaluation (M&E) information but also with the context information about the evaluation project. When organizations record the collected data from M&E projects, the context information is very often neglected. This can jeopardize the validity of comparisons among similar evaluation projects. We highlight this concern by introducing a quality in use assessment scenario. Then, we propose a solution by representing the context information as a new add-in to the INCAMI M&E framework. Finally, we show how context information can improve Web application evaluations, particularly, data analysis and recommendation processes.

Keywords: Context Information, Web Application, Quality in use, Measurement, Evaluation, INCAMI.

1 Introduction

While users are becoming more and more mature in the use of Web applications (WebApps), there are greater demands for the quality of these applications that match real user needs in actual working environments. Quality has therefore become a key success factor for organizations developing WebApps. Particularly, what definitively matters is the end user view of a WebApp's quality, namely, the quality in use. This means, as for [2], the capability of a WebApp to enable specific users to achieve specified task goals with effectiveness, productivity, safety and satisfaction in specified contexts of use.

The quality in use of a WebApp is affected by internal and external quality features such as usability, functionality, reliability, content, and efficiency, but it is also influenced by two resource components of its context of use: (i) the infrastructure, i.e. the computer, the network, or even the physical working medium; and, (ii) the user-oriented goals, i.e. the supported application tasks and the properties of the user type, such as level of training, expertise, and cultural

issues as well [3]. Note that tasks are the steps or sub-goals undertaken to reach an intended goal.

As remarked by Bevan, care should be taken when generalizing the results of any quality in use assessment to another context of use with different types of users, tasks, or environments [1]. Consequently, when designing and documenting well-specified *quality in use* M&E processes, at least the following information is needed [3]: (i) Quality in use attributes, metrics, and indicators, i.e. the data and metadata resulting from the design and execution of M&E processes, for a given information need; (ii) Descriptions of the components of the context of use including user type, equipment, environment, and WebApp tasks to be performed by users.

Therefore, if quality in use requirements, metrics, indicators, and context information could be thoroughly and consistently specified and recorded among different Web projects, the analysis and comparison of results should be sounder. Moreover, if we can provide all the above information, the external validity resulting from quality in use evaluations can be made stronger, then giving better support to the Bevan's recommendation.

In our previous line of research, support to M&E data and metadata was provided by the conceptual framework so-called INCAMI –which stands for Information Need, Concept model, Attribute, Metric and Indicator [10]. In this framework, measurement of attributes is specified by means of metrics, and evaluation by means of indicators. Additionally, a case study for quality in use evaluation was run on an e-Learning WebApp [3] but without specifying the context information consistently. In the present work, we discuss how a context information approach can be added to the INCAMI framework in order to strengthen it.

The main contributions of this paper are the following: we propose to include context information to quality evaluation of Web projects as a way to improve the data analysis and recommendations; we show how to add context information in the processes of specifying non-functional requirements, measuring and evaluating; and we give clues how context information can improve evaluation of a WebApp on a particular motivating scenario, i.e. in a quality in

use evaluation for an e-Learning WebApp (e-LApp).

The rest of this paper is organized as follows: In Section 2, a motivating scenario is presented to show the need of context information for M&E processes. The INCAMI framework is outlined in Section 3. Then, in Section 4, the approach to add context information to the INCAMI framework is discussed in the light of the motivating scenario. Next, Section 5 analyses related work; and finally Section 6 presents our conclusions and future work.

2 Motivating Scenario

In order to illustrate an actual evaluation problem and its possible solution, we describe here a scenario for quality in use assessment. Thus, we will put in evidence that, when designing M&E activities, the lack of a clearly structured context information specification can jeopardize the validity of conclusions drawn from analyzing quality in use assessment results. The motivating scenario is extracted from a recently performed case study of quality in use evaluation for an e-LApp [3].

Let's say we want to understand the quality in use for a particular e-LApp used in an introductory course of mathematics for pre-enrolled students at our School. A quality in use assessment is to be performed on this application based on agreed non-functional requirements, metrics, and indicators. Fig. 1 shows the quality in use instantiated model used for the case study, which is specified by characteristics and attributes. For instance, the "*Productivity*" characteristic is defined as "*the capability of the software product to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use*" [2]. The most commonly used resource is the time spent by users in performing tasks. In addition, one of its associated attributes is named "*Efficiency related to Task Completeness*", which is defined as "*the productivity of given users to perform the tasks completely*". On the other hand, the metrics and indicators were designed accordingly for these requirements following to some extent what were recommended for quality in use assessments in Section 1. Refer to [3] for further information.

In order to gather the proper data from the assessed e-LApp, a user test was designed and performed. The design included the definition of the user profile as well as the actual tasks to be performed by specific users. Four tasks were considered, e.g. "*browsing a specific content for the math course*" was one of them. Also, the environment on which the testing is performed was prepared. This included the physical space where users were working as well as the computing facilities needed to run the application, i.e. client and server machines. The user test and the assessment were replicated later on to strengthen the results.

Once the first user test was carried out and the M&E pro-

1. Quality in Use

1.1 Effectiveness

1.1.1 Task Effectiveness (TE)

1.1.2 Task Completeness (TC)

1.2 Productivity

1.2.1 Efficiency related to Task Effectiveness (ETE)

1.2.2 Efficiency related to Task Completeness (ETC)

1.3 Satisfaction

Figure 1. The instantiated model used to assess the quality in use of e-LApp.

cesses were executed, the degree to which the e-LApp satisfied the stated requirements was obtained, resulting in a value of 57.45 % [3]. Some weeks later, the user test on the same e-LApp was replicated. New users were carefully selected according to the same user profile, and the same four task descriptions and the instantiated quality in use model were used, along with the same environment settings. Surprisingly, the outcome was 74.70% to this second case.

In fact, that big difference between both results cannot be explained by differences in the user groups, since they were carefully selected according to the same user profile likewise the rest of the test. After analyzing the results both from user tests and evaluations, it turned out that, in the first evaluation, values corresponding to "*Productivity*" were much lower than in the second evaluation. Reviewing the measurement records, it was evident that both tests were carried out at different time frames of the day, in which the local network bandwidth in our School used to differ significantly, as it was checked in the log files. In the time frame of the second case the local network bandwidth reached values twice to three times higher than those that occurred in the first one. Then, the e-LApp responses in the first test were much lower than in the second test, affecting therefore the users' productivity. As a matter of fact, if we had clearly specified the local network bandwidth as a context property, together with M&E specifications, we would have recognized in a more straightforward way the differences between both assessment results.

As remarked above, differences in the context of use may affect effectiveness, productivity, and satisfaction, and, in the end, may risk the validity of comparisons among similar quality in use assessments. We can say then that the entity being assessed is *context sensitive* regarding M&E goals. Consequently, the relevant context information (e.g. the *local network bandwidth* attribute, among others) of the entity under analysis (i.e., a concrete e-LApp) should be clearly and consistently specified, together with M&E metadata and data, in order to achieve sounder analysis and recommendations.

3 Background

When dealing with assessments, organizations may benefit by specifying and storing M&E data and metadata in a structured way. With this aim, the INCAMI framework is based on a set of ontological concepts involved in the M&E activities [10]. These concepts are mainly *Information Need*, *Entity*, *Concept Model*, *Attribute*, *Metric*, *Measurement*, *Indicator*, *Evaluation*, *Measure* and *Indicator value*.

When we talk about data, we are referring to measure or indicator values yielded after a M&E process, e.g. 74.70 was the indicator value above. On the other hand, metadata means data about data. For instance, it can be the unit of a metric or indicator (e.g. %, link, second, etc.), the scale type (e.g. interval, ratio, etc.), among others. INCAMI assures that data always has associated metadata accordingly. This is also supported by an organizational catalogue, which allows not only the reusing of requirement, metric and indicator specifications but also the checking for consistency between outcomes from different organization's M&E projects.

INCAMI is a purpose-oriented framework and is made up of four main components (see [10] for more details):

The non-functional requirements definition and specification module, which deals with the definition of the *information need* for a particular *entity category*, and the specification of non-functional requirements by means of one or more *calculable concepts* and *attributes* related by a *concept model* –see Fig. 2. In our above scenario, the *information need* was “understand the quality in use of a concrete e-LApp that supports course tasks for pre-enrolled students”, where the *purpose* is “understand”; the *user viewpoint*, “pre-enrolled students”; further, the *focus* is on the “quality in use” *calculable concept*, and its *sub-concepts* like “effectiveness”, “productivity”, etc. In the end, these are related and *represented* by a “quality in use model”, in which an instance *combines attributes* as depicted in Fig. 1.

The measurement design and execution module, which deals with the specification of a concrete *entity* to be measured, the *metrics* which quantify attributes and the recording of the gathered *measures* after performing *measurements*. This module is centered on the metric concept. For example, the concrete e-LApp entity was “QPlus Virtual Campus”. In addition, the *indirect metric* named “Average of Task Completeness Ratio” quantifies the “Task Completeness” attribute (see [3]).

The evaluation design and execution module, which deals with the definition of *elementary* and *global indicators*, *decision criteria* and *elementary* and *global models*. Indicators interpret the level in which specified non-functional requirements has been achieved (see [3]).

Despite the INCAMI framework is well defined and structured, it lacks a key module when specifying and

recording, for instance, quality in use evaluations. That is to say, a well-developed module for context information is missing. INCAMI currently supports the specification of context information in the *contextInfo* field (see the *InformationNeed* class in Fig. 2), however, it is actually a loose and subjective way to do it.

4 Contextualizing M&E Projects

From the above motivating scenario, the need of tackling the context information concern for M&E projects –at least for quality in use assessments- was addressed. In addition, we highlighted INCAMI lacks a robust support for this aim. Therefore, in sub-section 4.1 we define what context information is. Particularly, we discuss a context information model and its integration to the INCAMI components. The new proposal ends up in a framework called C-INCAMI (that stands for *Contextual INCAMI*). In sub-section 4.2, we illustrate how context information is specified, and how can be used to fulfill the contributions stated in Section 1.

4.1 Context Definition and Modeling

4.1.1 Defining Context. To model context, first, we must understand what context is about [4]. To this aim we have analyzed some definitions to build a characterization of the *context* term that satisfies our needs. Next, we briefly present some definitions and views of context.

Dey [4] states that “*context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves*”. Gong proposes in [5] a quantifiable definition of context: “*the collection of an object and its relationships with other objects, including these objects*”. Strang et al [12] define the term context information as any information that can be used to characterize the state of an entity, i.e. a person, place or object, concerning a specific aspect –a classification- whose states (or instances) are expressed in a determined scale. Also they define an entity and a relevant aspect based on tasks and a state of such aspect. Kaltz et al [7] consider that context is determined by a set of structured context factors according to a context model, an application domain ontology, as well as a current perspective of such factors (the context information itself).

Once the previous definitions and approaches were analysed, we extracted a set of elements that allows us to characterize context information: (i) it is relative to or references a specific object or entity; (ii) the entity to which it refers should be relevant with regard to the task for which it is involved; (iii) it corresponds to internal or external entity properties and their relationships, under a specific situation;

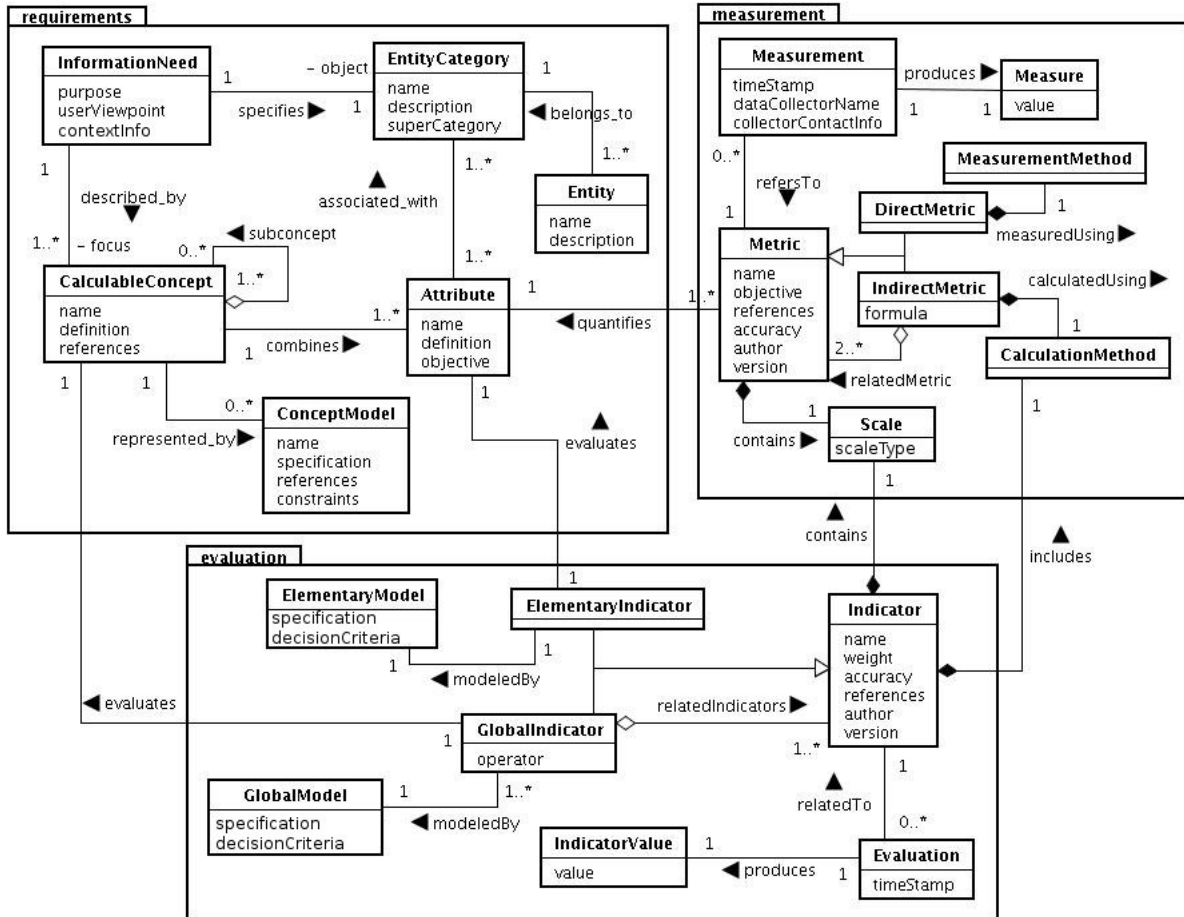


Figure 2. The main components and key terms of the INCAMI framework.

(iv) it should be relevant information with regard to (a) the specific task related to that entity, (b) the specific purpose related to that task, (c) the relevant properties of the entity (i.e. those involved in the task for which the entity is considered), and (d) the situation of the entity with regard to the task, and its interaction with other context elements; (v) it corresponds to a set of structured contextual factors (aspects) and properties, including the set of relationships between them, and an associated lexicon that makes their interpretation explicit.

As for INCAMI, the relevant entity for which context will be specified is the entity subject to M&E activities satisfying a specific information need (see *Entity* and *Entity-Category* classes in Fig. 2). For example, it can be the particular e-LApp in the motivating scenario, or any particular entity of a product, resource, process, or service. These entities are part of a project performed in an organization. Therefore, the relevant context information will be that affecting the stated information need for the entity and par-

ticularly its calculable concept, related to a) the project to which the entity is related to; b) the processes applied to the entity; c) the organization that runs the project itself, and; d) possibly, the environment external to that organization. For instance, the “local network bandwidth” is part of the relevant context related to the resources available for the project in which e-LApp entity is included, since it affects the application’s quality in use, and in turn, its productivity.

4.1.2 A Context Model for INCAMI. A context model is a key element in the development of systems following a context information approach [12]. The chosen representation for such a model should match the intended purpose for the context information taking also into account the trade-off between acceptable complexity and needed expressivity. In this sense, different representation approaches as those presented in [11] have been analysed in the light of a set of requirements of interest to our proposal, some of which are common to those identified in that work for ubiquitous

computing approaches: (i) context information should be able to be validated against a context model both at structure and instance level; (ii) contextual information should be precise and unambiguous allowing the coherent and consistent reuse of it; (iii) the context model should be able to be applied to current systems; in our case, to the current environment supporting the INCAMI framework; (iv) the model should be simple enough to be specified and processed with the aim of maintaining the system cost-performance trade off; (v) the model should be domain specific since it has to be focused on the M&E activities; (vi) the model should be flexible enough to allow its adaptation based on specific organization needs. Next, we describe the designed context model itself, and then, how it is integrated to INCAMI.

Regarding the context model itself, context is represented as an aggregation of *context properties* –as shown in Fig. 3. A context property is a specialization of the *Attribute* class (from the INCAMI requirement component). Hence, its semantic can be fairly defined (by means of a set of metadata), as well as the types of its values by means of a metric (recall Fig. 2), in order to avoid ambiguities for analysis, interpretation and comparison purposes. “*Local Network Bandwidth*” is, for example, a context property name, which is quantified by a designed metric using a numerical scale with a *kb/s* unit and a measurement method underlying in the *netperf*¹ tool. On the other hand, by means of the *relatedContextProperty* association it is also possible to establish relationships among any number of context properties, in order to reflect organization-specific domain issues. Related context properties may help discovering additional relevant properties as well. Following the previous example, “*Time Frame*” is a context property related to “*Local Network Bandwidth*” since the latter may change at different time frames of the day. Moreover, each property has an associated *context property type* whose semantic can also be defined. The types are mapped to key concepts to the software and Web engineering domain. In addition to this, and following the Kaltz et al [7] approach, context property types can be arranged in a taxonomy by using the *subTypeOf* association. This way, the types and its taxonomy can be defined and designed according to the organization needs. The idea behind defining context properties and context property types is to allow an organization to use its own software process ontology to describe context when designing M&E activities. Therefore, these activities are tailored to the organization and not vice versa. For example, the “*Local Network Bandwidth*” property is of the “*Resource Property*” type –where “*Software Project Property*” is its super type. These and other context property types were shown in Molina et al [8].

Regarding the context integration to the INCAMI framework we decided to keep two different information spaces,

¹<http://www.netperf.org/netperf/>

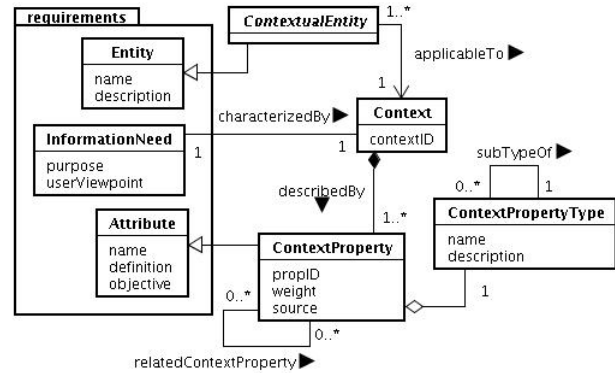


Figure 3. The model to add context information to the INCAMI framework.

namely: (i) *the space of domain elements*, i.e. the current INCAMI framework components as described in Section 3; and (ii) *the space of context elements*, as presented above. In doing this, the contextualization of the domain elements is achieved by establishing relationships with elements from the context space [7]. Thus, to describe the actual context of the entity being measured and evaluated in a specific project, the *Information Need* class maintains an association with the *Context* class (see Fig. 3). Besides, other elements of the INCAMI framework were found to be context sensitive, particularly, the concept model, metric, and elementary and global indicators. Their applicability in a M&E project may depend in turn on the context specified for the latter –e.g. for a more effective and coherent metric selection process at measurement design time. These elements were called *contextual entities* (see [8] for further details).

Lastly, to facilitate an organization-wide integration of context information, a common context catalogue may be used –as made with the metric and indicator catalogue [9]. Thus, context property types as well as definitions of context properties can be incorporated and stored by means of a collaborative reviewing system, as explained in [8]. Hence, each particular project in the organization will be using agreed context properties relevant to its specific needs, therefore, keeping the consistency with the rest of the organization’s projects both at data and metadata levels.

The resulting C-INCAMI framework is made up of the following modules –packages: (i) the *incami.requirements*, *incami.measurement*, and *incami.evaluation* modules, which are the former INCAMI modules (as presented in Section 3), and (ii) the *incami.context* module, which contains the concepts that define the basic structure for context, as well as those relationships that define its interaction with the INCAMI modules (see Fig. 4).

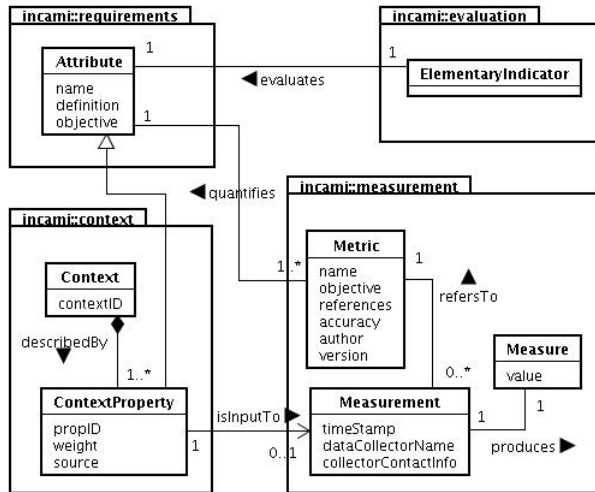


Figure 4. Integration of modules in C-INCAMI.

Summarizing, the proposed approach is simple, specific, flexible, semantically defined and able to be validated, organization oriented and loosely coupled, as discussed in [8].

4.2 Using Context Information

Here we show how context is specified together with M&E data and metadata, and how can be used to fulfil the stated contributions.

Let’s consider again the scenario from Section 2. As made before with INCAMI, we start by specifying the information need that guides the whole assessment process and, against which, outcomes are analysed. This includes defining the purpose, the entity, and the concept model that represents the focus concept –i.e. the quality in use. Following the new proposal, the relevant *actual context* is also specified as part of the information need (Fig. 3).

In order to do this, the evaluator selects from the organization’s context catalogue the relevant properties. For example, the context for the case includes a context property named “*Local Network Bandwidth*”, defined as “*The available bandwidth in the local network in a specific time frame*”. Other properties like “*Time Frame*”, “*Web Browser*”, and “*User Profile*” are included as part of the actual context. The values that actually describe the context are obtained by using a *Metric* associated to each property (see Fig. 4) likewise attributes are quantified in a concept model during the measurement implementation stage. Context properties’ values are specified by means of a *Measurement* object which in turn produces the corresponding value using a *Measure* instance. Specifically, the “*Local Network Bandwidth*” attribute is quantified by the “*Amount of Band-*

width Available” metric, which includes metadata like the measurement method, the scale, the unit, and so forth.

Once the context was designed and context properties’ values were collected, in order to assess the focus quality in use model –like in INCAMI- metrics and indicators are selected to quantify and interpret the attributes and characteristics accordingly. Also, user profiles and tasks are designed as discussed in Section 2. Finally, M&E are performed obtaining the final values.

Ultimately, from now on, not only M&E data and metadata are stored –as made before with INCAMI- but also the associated context data and metadata are stored as well. All this information can be used afterwards in the interpretation of assessment outcomes, allowing to improve undoubtedly the quality of analysis and recommendation processes. Recalling both results in the motivating scenario, if we had clearly specified the local network bandwidth as a context property, together with M&E specifications, we would have recognized in a more straightforward way the differences between both assessment results. Particularly, after looking at a simple and automated analysis report we would have identified the context differences and, then, concluding that both studies were not so closely comparable.

In short, following the C-INCAMI approach, if meaningful and sufficient context information is specified for M&E projects, we can assure that we won’t be “comparing apples and oranges”; thus, we can improve evaluations.

5 Discussion and Related Work

In this paper, we have shown how to add *context information* to an existing domain-specific framework, i.e. to INCAMI. It is worth mentioning that this approach is closely related to the so-called *context-aware* approach, which started to gain importance to the ubiquitous computing field, as well as recently to mobile devices as a smart solution to offer better services. For example, for applications like tourist or museum guides, a mobile user is the target entity, and the most relevant context aspects are time, location, user preferences, among others. However, we would like to make a distinction between our *context information* approach and the context-aware one as applied to the ubiquitous computing field. In the latter, the term aware gains more relevance in the sense that the actual context is captured by the application actively, i.e. it is continuously sensed as it changes. This raises additional issues than those covered by our proposal. Hence, we have chosen to distinguish the approach applied here by calling it *context information approach*. On the other hand, regardless the above distinction, and taking into account different context-aware applications [4, 5, 7, 11, 12], as well as the definition of the term aware² as “*having knowledge or perception of a*

²From Ask Oxford Website

situation or fact”, we believe the context-aware approach could be therefore applied to any field in which the relevant context information is necessary and useful to offer basically better services to its users. In this way, the application will be built regarding the application domain, the relevant properties of the context (e.g. position, time, or more specific ones), and current requirements. As an example, in [6] the context-aware approach is applied to knowledge management in business environments to ease and improve the use of organizational information. Nevertheless, both approaches possess the same challenge when dealing with context representation in a machine-readable fashion and its integration into existing applications. For instance, the simplicity, flexibility and expressiveness requirements are key issues. In our approach, a context model that is both simple and domain specific has been achieved, with such a level of semantic expressiveness and flexibility that can be fitted to diverse software and Web organization’s information needs with regard to M&E processes. Moreover, the context model can be applied to different domains and reused to existing applications due to its loose coupling with the INCAMI framework’s components.

6 Conclusions and Future Work

In this paper, we have discussed the C-INCAMI approach, and particularly, how the INCAMI framework can be strengthened by adding and integrating to it the context component. From now on, not only M&E data and metadata are stored –as so far performed with INCAMI–, but also the associated context information data and metadata are recorded as well. In this way, a more comparable and robust data and information processing can be made, then potentially improving the quality of analysis and recommendation processes.

To illustrate the need and usefulness of our proposal, a motivating scenario based on a previous study of quality in use assessment has been considered, in which the importance of specifying context information was highlighted. Specifically, it became clear that differences in the context of use may risk the validity of comparisons among similar quality in use assessments.

Lastly, C-INCAMI offers not only a simple, flexible, goal-oriented organization approach for M&E programs but also a consistent, semantically defined and able to be validated mechanism for representing context information. By including in decision-making the relevant context information that bears on the interpretation of M&E results, we hope ultimately to make the quality of key entities in the organization more visible, consistent and comparable .

An ongoing research is the implementation of the M&E organizational memory system integrated to the contextual INCAMI framework and tool.

References

- [1] Bevan N. Quality in use: meeting user needs for quality. *Journal of Systems and Software*, 49(1), pp. 89-96, 1999.
- [2] ISO/IEC 9126-1. “Software Engineering - Product Quality - Part 1: Quality Model”, 2001.
- [3] Covella G. Olsina L; Assessing Quality in Use in a Consistent Way, In proc. of ACM, Int’l Congress on Web Engineering, (ICWE05), SF, USA, pp.1-8, 2006.
- [4] Dey A.K.: Understanding and Using Context. *Personal and Ubiquitous Computing Journal (Online)*, Springer London, Vol. 5 (1), 2001, pp. 4-7.
- [5] Gong L.: Contextual Modeling and Applications, *IEEE International Conference on Systems, Man and Cybernetics*, Vol. 1, pp. 381-386, 2005.
- [6] Huang W., Tao T.: Adding Context-awareness to Knowledge Management in Modern Enterprises. 2nd IEEE Int’l Conference on Intelligent Systems, 2004.
- [7] Kaltz J.W., Ziegler J., Lohmann S.: Context-aware Web Engineering: Modeling and Applications. In: *RIA - Revue d’Intelligence Artificielle, Special Issue on Applying Context-Management*, Vol. 19 (3): pp. 439-458, Lavoisier, Paris, France, 2005.
- [8] Molina H; Olsina L.; Towards the Support of Contextual Information to a Measurement and Evaluation Framework, In *IEEE CS Press, 6th Int’l Conference on the Quality of Information and Comm. Technology (QUATIC’07) Portugal*, pp. 154-163. 2007.
- [9] Molina H; Papa F; Martin M; Olsina L: Semantic Capabilities for the Metrics and Indicators Cataloging Web System. *Engineering Advanced WebApps; Matera, Comai Eds; Rinton Press, US*, pp. 97-109, 2004.
- [10] Olsina L; Papa F; Molina H.: How to Measure and Evaluate Web Applications in a Consistent Way, Ch. 13 in Springer Book “Web Engineering: Modelling and Implementing Web Applications”; pp. 385-420. Rossi, Pastor, Schwabe, and Olsina Eds., 2007.
- [11] Strang T; Linnhoff-Popien C: A Context Modeling Survey. *Workshop on Advanced Context Modelling, Reasoning and Management, UbiComp’04; 6th Int’l Conf. on Ubiquitous Computing, UK*, pp.34-41, 2004.
- [12] Strang T; Linnhoff-Popien C; Frank K: CoOL: A Context Ontology Language to enable Contextual Interoperability. *DAIS’03, LNCS 2893*, pp. 236-247, 2003.