Quality Models for Web [2.0] Sites: a Methodological Approach and a Proposal

Roberto Polillo[1]

[1] University of Milano Bicocca, Dept. of Informatics, Systems and Communication, Building U14, Viale Sarca 336, 20126 Milano (Italy)
roberto.polillo@unimib.it

Abstract. This paper discusses a methodological approach to define quality models (QM) for Web sites of any kind, including Web 2.0 sites. The approach stresses the practical use of a QM, in requirement definition and quality assessment, during design & development processes or during site operation. An important requirement for such QMs is *organization mapping*, which allows who is in charge of quality management to easily identify the actors in the organization responsible for implementing or improving each specific quality characteristic. A family of QMs is proposed and compared with ISO/IEC 25010 QMs for software products and software-intensive computer systems.

Keywords: quality, quality model, web, web engineering, web 2.0, ISO/IEC 25010

1 Introduction

According to ISO/IEC 25000:2005 [1], a *quality model* (QM) is a "defined set of characteristics, and of relationships between them, which provides a framework for specifying quality requirements and evaluating quality."

QMs are very important in Web engineering. Having a good QM at hand can be extremely useful in all phases of a Web site life cycle. In the requirement specification phase, a QM helps in elicitating and orderly describing all important facets of the site to be designed. Indeed, the table of contents of a good requirement specification document could strictly mirror the QM, by assigning to each model characteristic a specific section of the document [2]. During the development process, a QM helps the project team in keeping their eyes on all desired quality attributes of the system to be implemented. In assessing the quality of an existing site, or different sites for comparison or benchmarking, a QM provides a structured approach to the evaluators, helping them to stay focused on the important issues. In the operation phase, a QM provides the site management with a "compass" to keep its evolution on the right track. Indeed, all Web sites are very dynamic; their evolution is constant and substantial: it is therefore essential to continuously monitor their quality, to avoid that the frequent changes disrupt piecemeal an initially sound project. This is particularly important for Web 2.0 sites, whose evolution is determined not only by the site

management, but also by the (possibly large and uncontrollable) user community. A "suitable" QM is the necessary supporting tool for these monitoring actions.

But *how* do we choose it? The selection of a QM is a delicate task, because it may have a large impact on the site's success, and is not trivial at all, for two main difficulties: *orthogonality* and *measurability* of characteristics. Orthogonality is difficult to achieve because the quality attributes of a Web site interact in complex ways; measurability, because many of them are subjective.

The literature on Web quality is very large, and a number of QMs for Web sites have been proposed over the years, approaching the problem from different perspectives. QM characteristics may be chosen on the basis of their semantic orthogonality, their measurability, the feasibility of their automatic evaluation, their relationship with the Web site development process, or with the use of statistic or probabilistic models (among others: [3],[4],[5],[6],[7]). Some QMs address specific types of Web sites, such as e-commerce or information portals; others analyze specific attributes, like data quality or quality in use (e.g. [8], [9], [10], [11]). Most of them are in some way related with the ISO quality standards. However, there seems to be no general consensus on their definition and characteristics.

This paper will contribute to this debate, by proposing an approach specifically oriented to the needs of the people responsible for the *management* of a Web site, and by sketching a QM family which can be proficiently used by project managers and Web properties managers both in the development and operation phase. This is a revision and extension of a simple QM for Web 1.0 sites previously defined by the author [4], following its experimentation in the Web site development road-map described by the author in [2] and the Web evolution of recent years.

In Section 2 the ISO approach to QMs for software and computer systems is summarized. Section 3 will discuss the main peculiarities of Web sites with respect to traditional software systems, and lay down a few basic requirements for Web sites QMs, also considering the evolution of the role of users in Web 2.0 sites. Section 4 will describe the proposed QM family, and Section 5 will briefly compare it with the ISO standard. Finally, Section 6 will contain some conclusions.

2 The ISO System and Software Quality Models

In the software engineering literature, software QMs have been discussed for many years. The ISO/IEC 9126, issued as an International Standard (IS) in 1991 [12] and revised in 2001 [13], is the best known reference in this area. Part 1 of this multi-part document provides a very general QM for software products external and internal quality, based on a set of 6 *quality characteristics* (*Functionality*, *Reliability*, *Usability*, *Efficiency*, *Maintainability*, *Portability*) and 27 *sub-characteristics*. A second QM defines 4 characteristics for *Quality in use, i.e.* "the user view of the product". This IS has been recently canceled, and replaced by ISO/IEC 25010 [14], which updates the previous QMs in various ways. It addresses "software products and software-intensive computer systems" of any kind, and defines two QMs. The *Product quality model* encompasses internal and external qualities of the system, and is composed of 8 characteristics and 31 sub-characteristics (Fig.1).

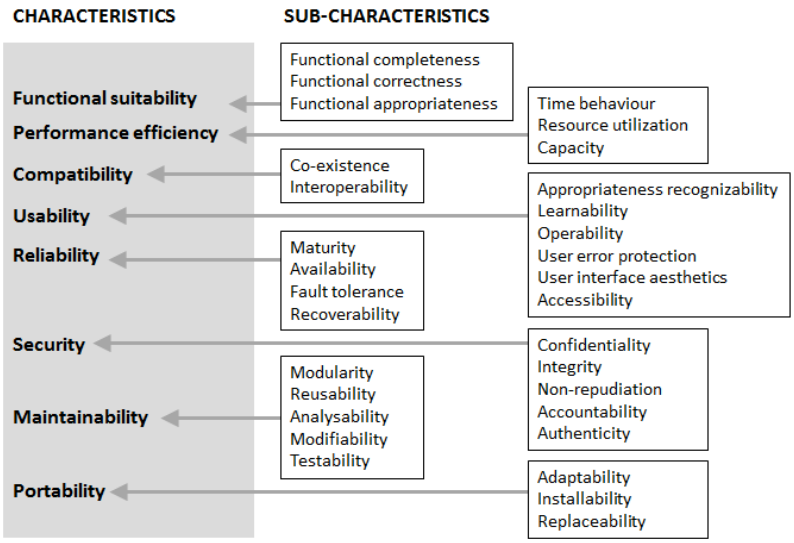

Fig. 1. Product quality model according to ISO/IEC 25010

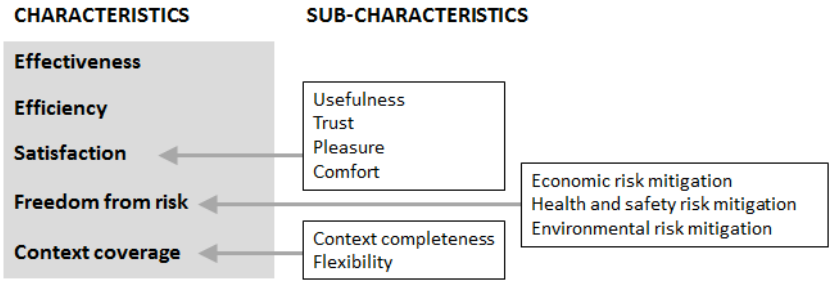

Fig. 2. Quality in use model according to ISO/IEC 25010

The *Quality in use model* is now composed of 5 characteristics and 9 sub-characteristics (Fig.2). Note that quality in use is a superset of usability, classically defined in [15] as "the degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use."

Each QM sub-characteristic may be further hierarchically decomposed. Quality characteristics and sub-characteristics at any level should be *measurable*, either directly or indirectly, through a set of associated *measurable properties*.

Fundamental in the ISO approach is the distinction between the *internal properties* of a product (which contribute to the *internal quality*), its *external properties* (which contribute to the *external quality*), and its *quality in use properties*, i.e. properties

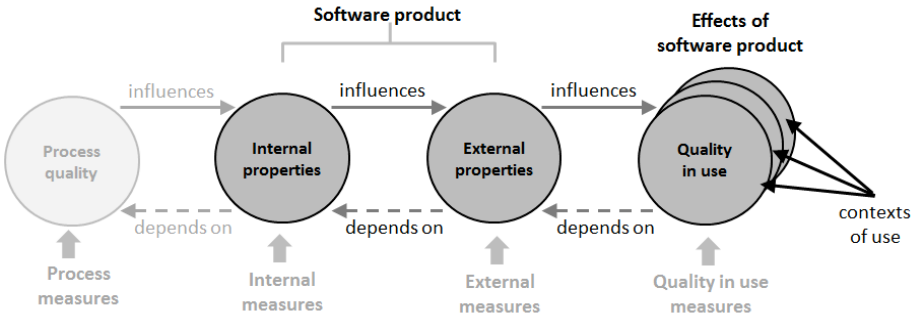

Fig. 3. Conceptual approach to quality, according to ISO/IEC 25010

which can be measured when the product is actually in use in specific contexts. All these properties influence each other and the resulting quality in a complex way, as schematized in Fig.3.

ISO/IEC 25010 belongs to the SQuaRE series of International Standards (see [1]). In SQuaRE, ISO/IEC 25012 [16] defines a third QM, for data retained in a structured format within a computer system, composed of 15 characteristics: *Accuracy*, *Completeness*, *Consistency*, *Credibility*, *Currentness*, *Accessibility*, *Compliance*, *Confidentiality*, *Efficiency*, *Precision*, *Traceability*, *Understandability*, *Availability*, *Portability* and *Recoverability*.

3 Quality Models for Web Sites: Why They Should be Different

3.1 Web Sites Peculiarities

The ISO standards provide a very general conceptual framework for defining QMs for complex systems with a substantial software component. The basic approach of defining a hierarchy of quality *characteristics*, and *measurable properties* which can be aggregated to obtain quantitative measures of characteristics provides a sound foundation for defining *any* QM, in *any* domain. Moreover, the ISO model is the result of three decades of discussions about the basic quality dimensions of software-based systems. Its categorization and terminology can be discussed and - in a few cases - may also be considered somehow obscure, but certainly cannot be ignored in any approach to quality in software engineering.

On the other hand, it should be clearly understood that the ISO documents only provide a *conceptual framework*, and not a ready-to-use QM. To be of practical use, this framework must be tailored to the specific [class of] system[s] under consideration. This may not be a simple task, especially when these systems do not fit well with the systems considered in classical software engineering, such as ERP, command & control, embedded systems. This is the case of Web sites, which possess a number of peculiarities that greatly differentiate them from the above systems:

Information content. In the large majority of cases, unstructured information content prevails on structured data. Emphasis is on user navigation, not on data management and computation. Therefore, a fundamental dimension of quality relates to *information architecture* [17]. Information architects are more and more involved in large Web sites, together with *content editors*, who create and manage its information content. Information-rich sites may employ large editing staffs, with an organization in some ways similar to that of traditional magazines.

Communication. In most cases, Web sites can be considered machines whose main purpose is communication, rather than computing and data management. This is also true for e-commerce or other sites offering online services. Web sites address a global audience, in a strongly competitive, "open" environment. There is no user lock-in: competition is only a few clicks away, so visitors' loyalty must be won on a day-by-day basis. User attention span can be extremely short, so his/her interest must be captured in brief time-intervals. So big efforts are required on communication and branding, and professionals typically not seen in traditional software projects are necessary (visual designers, art directors, communication and marketing people).

Continuous evolution. Web sites are living organisms. Their contents are constantly updated, and even their information architecture changes frequently. This is true for *any* site, not only for information portals. Visitors of a site often expect the content to be updated practically in real time. Site managers must strive hard to comply with these expectations, just to keep their site reputation. Interactive services and the user interface are frequently modified and improved. According to the *perpetual-β* concept, the software behind these services is continuously modified to better serve user needs. These – in turn – change as new possibilities are discovered, in a constant *co-evolution* of usage patterns and system functions. In a word, managing the evolution of a Web site sets pressing requirements to site administrators, and this should be taken into account seriously in any QM designed for these systems.

3.2 Web Site Quality Actors

By [*quality*] *actor* we mean any system stakeholder with an active role in creating/maintaining some quality attribute, such as Web designers, visual designers, content editors, software developers. Actors of a Web site are more numerous and more varied than in traditional software systems. Indeed, the development of any site is really a multi-disciplinary project, involving many different roles (Fig.4).[1]
In a typical Web 1.0 site, end users have a passive role, so they are not considered actors because they do not contribute to its quality: they only navigate the site and possibly interact with it in predefined transactions (as in e-commerce). In Web 2.0 sites the situation is completely different. The users can typically create and upload content, embed content from other sites, tag, comment or rate content created by other

[1] Different roles may not necessarily be played by different people. For very small sites, all of the above roles may also be impersonated by the same person.

users and share it with their "friends", and interact with them in public. This is not only true for large social networks such as Facebook, Twitter, YouTube and Flickr, but also for an increasingly large number of small sites, due to the many available

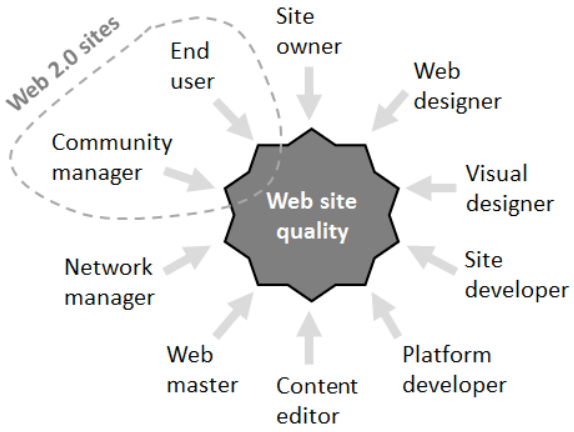

Fig.4. The main quality actors of a Web site

tools which allow to easily implement these functions, such as share buttons, plugins, html snippets. Therefore, in Web 2.0 sites, *the users themselves must be considered quality actors* and critical ones indeed, since they can have a big impact on the global functioning of the site. Even a perfectly designed and implemented site can fail as a consequence of "bad" (or unexpected) user behavior. So users must be continuously monitored and in some way controlled or stimulated, requiring the presence of new roles (denoted as *community management* in Fig.4), and in some cases the evolutionary modification of specific site functions, intended – so to speak – to improve the user-generated quality. A typical example is the evolution of the community content moderation mechanisms in Yahoo!Answer, where they had to oppose the unexpected volume of user spam and troll activity, that seriously risked crashing the site [18].

3.3 Organization Mapping

The ISO definition of a QM, quoted in Section 1, emphasizes the practical purposes of any QM, which is not viewed as a mere categorization of the quality attributes of a system, but rather as a *practical tool*, to steer design ("specifying requirements") and evaluation ("evaluating quality") processes. In our view, this should be constantly kept in mind when defining any QM. To this end, we require that there be as simple as possible relation between quality [sub-]characteristics and the roles (actors) responsible for implementing and improving them. In this way, responsibility for different quality characteristics can be easily allocated and tracked, being always clear who is responsible for what. We call this attribute of a QM *organization mapping*. In Fig. 5, mapping on the left can be considered better than the mapping on the right,

because responsibilities are better isolated and quality characteristics improvements are easier to manage.

A good mapping is a crucial requirement of a Web site QM because, as shown in Fig.4, the actors involved in Web projects are many, and the involved skills are extremely varied. In a multi-disciplinary team, different cultures, practices and value systems may sometimes create interaction difficulties, as anybody involved in medium to large Web site development or operations may have experienced. To avoid these problems, it is necessary that the teams be correctly organized, with a clear allocation of responsibilities on the different system components and associated quality characteristics.

Of course, the goodness of the mapping does not depend only on the QM, but also on the actual organization which develops and manages the site. A chaotic organization will nullify the practical utility of even the best QM. Nevertheless, after fifteen years of Web engineering experiences, the roles and functions of the different quality actors in e Web project are today sufficiently well understood. This allows to define good QMs which are reasonably applicable to most Web organizations.

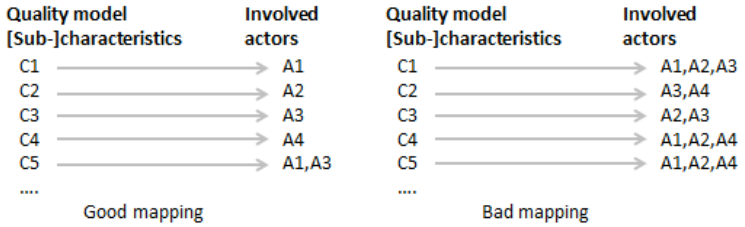

Fig. 5. Organization mapping of quality models

3.4 Requirements for Web Sites Quality Models

With the above premises, we can now lay down the main requirements for our QM.

Requirement 1. QM should have an organization mapping as simple as possible, as discussed in 3.3. We do not require that it be related to a *specific* project organization schema, but simply that the quality [sub-]characteristics be associated in a simple way to the quality actors of Fig.4. It is rather evident that the ISO QM of Fig.1 does not satisfy at all this requirement.

Requirement 2. QM should be tailorable to the class of sites under consideration. Web sites are enormously diversified. They may differ in size, in technology, in purpose, in complexity, in relationship with the front users (from purely informative to interactive to social), in impact on their activities (from critical to non-critical). So there will be no universal QM. Tailoring the QM would mean adding or dropping some sub-characteristics or specializing some of them with further levels of detail. Sometimes we would also assign *different weights* to the [sub-]characteristics, to express their importance in the particular context.

Requirement 3. QM should be subsettable according to its specific purpose. Some [sub-]characteristics should be droppable from the QM, when they are not needed in its actual context of use. E.g., when using a QM to compare a site with its competition, we usually do not have access to information on their internal structure. Thus, we would drop all [sub-]characteristics associated to internal properties from the QM.

Requirement 4. QM should be scalable according to site complexity. Any site (even the simplest) is really a very complicated system, as briefly discussed in 3.1. But it is totally unrealistic to pretend that small organizations (which own the large majority of sites) may (or want to) deal with all the subtleties of a conceptually sound and complete QM. Simple users need simple tools. Therefore, a scalable QM would be available in simplified versions to be used in simple contexts.

Requirement 5. QM should be universally usable and accessible. Last but not least, if we want to have a real impact on the quality of the present day Web, we should design QMs that, as much as possible, are *universally usable and accessible*. This would entail the use of broadly understood concepts described in a simple language, with easy and free accessibility.[2]

The stated requirements imply that we need a *family of closely related QMs*, and not a single QM, if possible with a common set of top-level characteristics. These are the "foundations" of the QM, and therefore should be easily recognizable by anybody as the basic dimensions of the quality of *any* Web site. They would constitute the main sections of the requirement specifications of *any* Web development project, and the main aspects to be considered in *any* assessment or evaluation. QM personalization should then be localized in the lower levels of the hierarchy of characteristics, to cope with specific Web applications (Req.2), purpose (Req.3), site complexity (Req.4) and to the complexity of the organization (Req.5). This will be mostly done by adding/dropping sub-characteristics or defining lower levels in the characteristics tree.

4 A Quality Model Family for Web Sites

4.1 Defining the Top-Level Characteristics

Rather than start from the ISO model and modify it piecemeal to comply with the stated requirements, it seems more reasonable to start anew, and see where this approach leads. Requirement 1 suggests to start by defining a general model of a Web

[2] Lack of usability and accessibility are, in our opinion, the main problems with the ISO QMs, which hinder their large scale adoption by the general community of Web practitioners. ISO documents are difficult to read and organized in a complex structure, which is continuously evolving. To understand the status of the ISO document system and to identify the documents relevant to a particular activity, it is not easy and very costly, since documents are not freely available, but cost a lot of money. Regrettably, this is also true for quality related standards, which should be, in our opinion, as openly available as possible.

site, showing its main logical components (the quality of which we wish to take under control), its main quality actors and the relationship between actors and components. This can be done a)- considering the Web site design & development process, or b)- considering the Web site in operation. The second approach seems more comprehensive because of the constantly evolving nature of Web sites (which are not "frozen" when they are published online after development) and because it allows to consider the role of end users as quality actors, which is fundamental in a Web 2.0 context. Thus this paper will use approach b).[3]

Therefore, a Web site in operation will be modeled as a set of nested logical components, as shown in Fig.6. The *Site* component is nested in a *Site platform*, typically representing the used Content Management System (CMS) and related software components (e.g., DBMS). In turn, the *Site platform* is nested in the *Server & Network Platforms* component, representing the server(s) hosting the site and the network infrastructure. The *Site* component is in turn decomposed in five components: *Information architecture & navigation*, *Graphics & branding*, *Software functions* and *Content* components, which are self-explaining.

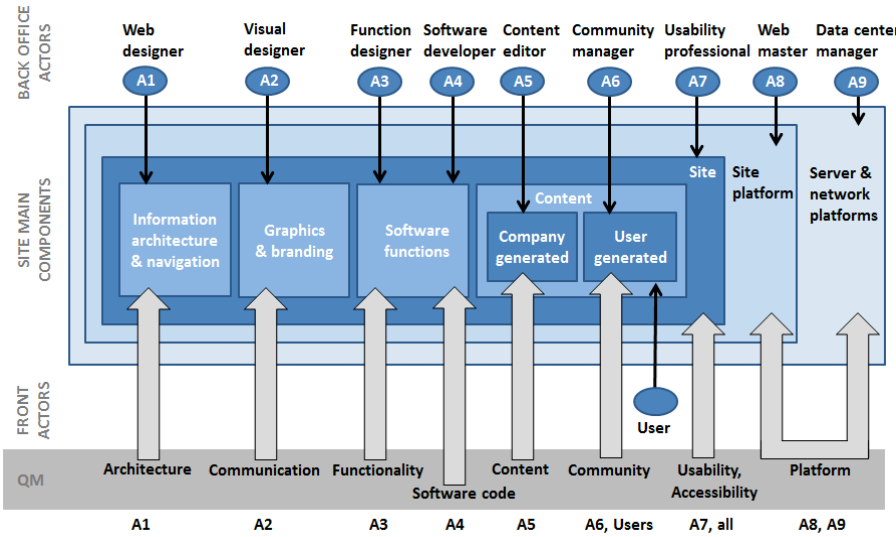

Fig. 6. A general model of Web site components and quality actors, and the resulting QM

Each logical component is associated to its (prevailing) quality actor. For example, the *Company generated content* component is under the responsibility of *Content editor(s)*. Actors are all members of the back-office organization, except in Web 2.0 sites, where the users are also considered actors. The bottom line in Fig.6 shows the 9

[3] Approach a) has been used in the previous version of this QM, dealing only with Web 1.0 sites [4], using the design & development model described in [2], in which the quality of a Web site is formed incrementally, through an ordered iterative process. Not unexpectedly, the results are the same, since the same actors and components are present in both approaches. Lack of space does not allow to further comment on this issue here.

top-level characteristics of the proposed QM: *Architecture*, *Communication*, *Functionality*, *Software code*, *Content*, *Community*, *Usability*, *Accessibility*, *Platform*.

Here, the term *Architecture* refers exclusively to information architecture [17], including site navigation facilities, and not to internal software architecture. Its associated actor is therefore the Web designer (or information architect).

Communication refers to all aspects of site communication, typically embodied in the site Style Guide, defining graphics, typography, multimedia usage and user experience issues. The associated actors are the visual designers because in small/medium sites this responsibility is usually assigned to them. Note, however, that larger sites may have a more complex organization, involving art directors, communication departments, and the like.

Like the ISO *Functional suitability* [14], *Functionality* means "the degree to which the site provides functions that meet stated and implied needs when used under specified conditions". Note that this does not include navigation functions (menus, breadcrumbs, and so on), which are part of the site *Architecture*.

Content collects all the quality characteristics related to the company-generated information/data content of the site, under the responsibility of the content editors.

Community is mostly used only for Web 2.0 sites, and considers user-generated content: associated actors are site users and site community managers.

Platform considers the site platform (CMS, DBMS, and similar components, under responsibility of the Web master), the hardware and software of the hosting servers, and the network infrastructure. Its quality characteristics are both static (i.e.: are they suitable for the context?) and dynamic (i.e.: are their operations well managed? Are their performances adequate?). Here the quality actors may differ depending on the specific organization: in Fig.6 we consider the case when server and network management are outsourced to an external organization, and there is a data center manager interfacing the service.

Usability and *Accessibility* have the usual meaning of the ISO documents. Since these characteristics are the result of the cooperation of all involved actors, in Fig.6 we have indicated a usability professional, as the actor with the responsibility of managing the usability and accessibility issues of the site.

Finally, *Software code* refers to the quality of the software specifically developed for the site (therefore excluding platform components acquired on the market), under responsibility of the software developers.

Note that in most cases there is a one-to-one relationship between characteristics and actors, as shown in the bottom line of the schema, thus the QM has a good *organization mapping*, as required.

Because the names chosen for the top-level characteristics are very mundane, the site quality profile can be easily communicated to *all* site stakeholders, e.g. with a simple radar diagram, as in Fig.7.

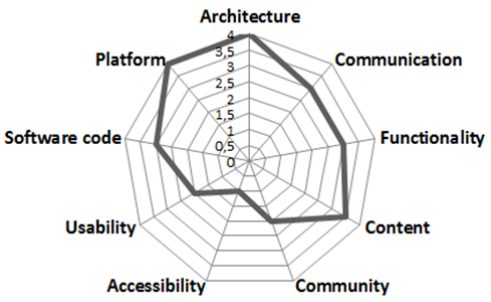

Fig. 7: The quality profile of a Web site

4.2 Defining the Sub-Characteristics

The definition of sub-characteristics is less critical. Once the top-level framework is stable and well understood, the lower levels can be tailored to specific contexts and improved over time, as experience in their use increases and Web applications evolve.[4] Our proposal is based on 33 sub-characteristics (Fig.8), including internal quality (*Standards conformance*, *Code* and *Platform Maintainability*) and Web 2.0 sites (*Community Management*).[5] These should be dropped when assessing only external quality and quality in use of Web 1.0 sites, as in [4].

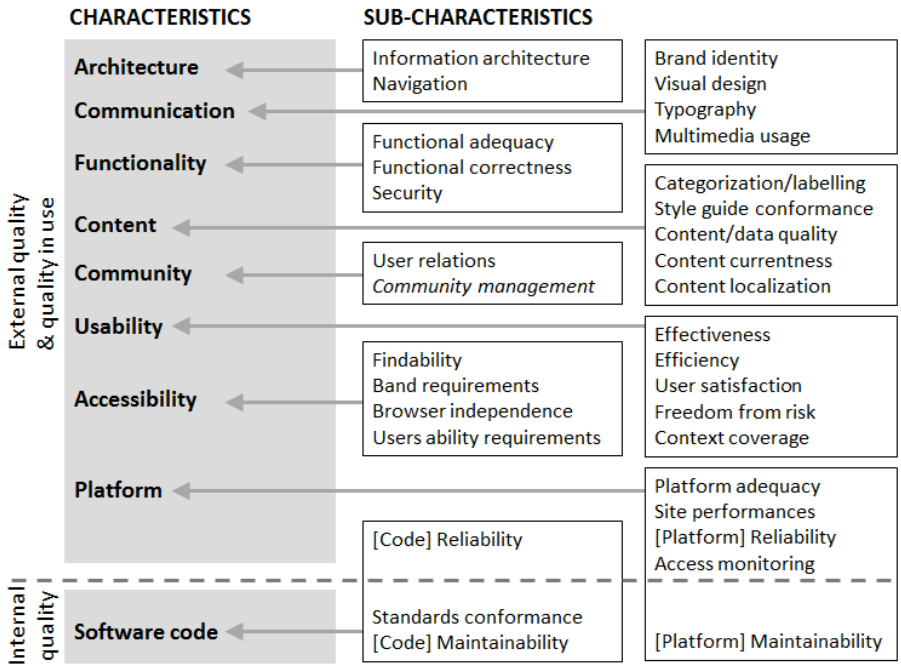

Fig. 8. The proposed QM. Sub-characteristics for Web 2.0 are in italics.

In most cases, the meaning of sub-characteristics is self-evident.

For *Content*, we evaluate separately the content categorization (*Categorization/labelling*),[6] the conformance to organization-defined style guides –

[4] The ISO standard itself explains that "the set of sub-characteristics associated with a characteristic have been selected to be representative of typical concerns without necessarily being exhaustive" (see [14], pag.2).

[5] *User relations* concern the management of user requests, and applies also to Web 1.0 sites.

[6] Note that, while the *Architecture* characteristic deals with the overall information architecture and navigation of the site, the *Categorization/labelling* sub-characteristic deals with the organization of the site contents. They are considered separately, because one is defined by the Web designer, the other by the content editors.

possibly including external standards (*Style guide conformance*), the timeliness of content updating (*Content currentness*), and the quality of internationalization and localization of the site (*Content localization*). Finally, *Content/data* quality would consider such attributes as accuracy, completeness, and so on of the content elements (including structured data stored in data bases), such.

Note that here *Accessibility* is intended in its wider meaning, as the characteristic that can limit the digital divide. Therefore, its sub-characteristics are: *Findability*, *Band requirements*, *Browser independence* and *User ability requirements*, i.e. accessibility for people with disabilities.

For *Usability*, we used, for simplicity, the characteristics of the ISO/IEC 95010 model for *Quality in use*.[7]

Under *Platform*, *Platform adequacy* collects all issues referring to the static properties of CMS, DBMS, server and network infrastructures (hardware and software), while *Site performances* deals with their dynamic properties (time behavior, resource utilization, and so on). *Access monitoring* evaluates SEO and Web analytics activities.

We considered *Reliability* and *Maintainability* separately for the site-specific *Software code*, and for the (often standard) *Platform,* since the involved quality actors are usually different.

A number of third-level characteristics should be further defined, tailored to specific classes of sites. This is typically the case of *Functional adequacy,* to deal with specific classes of functionalities, such as *Shopping functions adequacy* for e-commerce, *Uploading functions adequacy* for file sharing sites, *Identity profiling functions adequacy* for social networks, and so on.

As we shall see in the following sections, a number of ISO/IEC 25010 sub-characteristics can be used at the third or fourth level of our QM. For *Content/data quality*, sub-characteristics of the ISO/IEC 25012 data quality model may be used, such as *Accuracy*, *Completeness*, *Consistency*, *Credibility*, *Precision*, *Traceability*, and so on.

5. Comparison with the ISO Standard

A comparison between ISO 25010 and the proposed QM shows the following main differences and analogies:

1. Top-level characteristics *Architecture*, *Communication*, *Content* and *Community* and their sub-characteristics, which differentiate Web sites from traditional software systems, are not considered in ISO/IEC 25010 models. (Structured data - but not textual and multimedia information - are considered in ISO/IEC 25012).
2. *Functionality* is included in both models (though with slightly different names). In our model, *Security* is a sub-characteristic of *Functionality* (as it was in

[7] Indeed, the ISO/IEC 25010 itself specifies that "Usability can either be specified or measured as a product quality characteristic in terms of its sub-characteristics, or specified or measured directly by measures that are a subset of quality in use." (§4.2.4) We prefer the second option, closer to the "classical" definition of usability [15].

ISO/IEC 9126:2001), but it might be considered a top-level characteristic as well if desired (this would be advisable, e.g. in Web banking applications).

3. *Usability* is included in both models. As sub-characteristics, we considered the characteristics of *Quality in use* (Fig.2).

Table 1. Comparison of ISO/IEC 25010 vs the proposed QM.

ISO/IEC 25010 Product QM	Corresponding level in the proposed QM
Functional suitability	1 (name changed to Functionality)
Functional appropriateness	3 (under Functional adequacy)
Functional completeness	3 (under Functional adequacy)
Functional correctness	2 (under Functionality)
Performance efficiency	Not used
Time behaviour	3 (under Site performance)
Resource utilization	3 (under Site performance)
Capacity	3 (under Platform adequacy)
Compatibility	3 (under Platform adequacy)
Co-existence	4 (under Compatibility)
Interoperability	4 (under Compatibility)
Usability	1 (uses sub-attributes of ISO Quality in use QM)
Appropriateness recognizability	Not used
Learnability	Not used
Operability	Not used
User error protection	Not used
User interface aesthetics	Not used
Accessibility	1 (top-level characteristic)
Reliability	2 (under Software code and Platform)
Maturity	3 (under Reliability)
Availability	3 (under Reliability)
Fault tolerance	3 (under Reliability)
Recoverability	3 (under Reliability)
Security	2 (under Functionality)
Confidentiality	3 (under Security)
Integrity	3 (under Security)
Non-repudiation	3 (under Security)
Accountability	3 (under Security)
Authenticity	3 (under Security)
Maintainability	2/3 (under Software code and Platform adequacy)
Modularity	3/4 (under Maintainability)
Reusability	3/4 (under Maintainability)
Analysability	3/4 (under Maintainability)
Modifiability	3/4 (under Maintainability)
Testability	3/4 (under Maintainability)
Portability	3 (under Platform adequacy)
Adaptability	4 (under Portability)
Installability	4 (under Portability)
Replaceability	4 (under Portability)

4. We put *Accessibility* at the top-level, given its importance in many Web sites (in ISO it is a sub-characteristic of *Usability*).
5. While *Maintainability*, *Portability* and *Compatibility* are given much emphasis in ISO/IEC 25010, they do not need a front-line position in present day Web sites, more and more built on-top of widely used and compatible platforms, in some cases maintained by large communities of developers. We considered *Maintainability* separately for the site-specific *Software code*, and for the (often standard) *Platform*, as a second level characteristic. *Compatibility* and *Portability* do not appear in Fig.8, as they may be considered third level characteristics under *Platform adequacy*, for the evaluation of the selected platform, and a component of *Browser independence*, under *Accessibility*.

In summary, with respect to ISO/IEC 25010, the proposed QM considers some new characteristics related to the Web sites specificities, has an higher level of abstraction and allocates common sub-characteristics in a different way, according to their level of importance in Web sites and to the organization mapping requirement.

A detailed mapping between ISO/IEC 25010 and the proposed QM is shown in Table 1. Here, the shaded [sub-]characteristics are not used in our QM, but might be added at third or fourth level in the hierarchy, where indicated in the table. If this is done, our QM can be said to conform to the ISO standard, being a superset of it.[8]

5 Conclusion

This paper has proposed a methodological approach to define QMs for Web sites of any kind, including Web 2.0 sites and applications. The approach stresses the practical use of a QM, in requirement definition and quality assessment, during design & development processes or during site operations. Therefore, the main driver for QM definition has been what we called *organization mapping*, as opposed to the conceptualization of abstract quality characteristics. Organization mapping allows who is in charge of quality management to easily identify the actors in the organization responsible for implementing or improving each specific quality characteristics. This is much more important for Web sites than in traditional software systems, given the high number and diversity of the actors involved, and the possibility of conflicts arising from their diverse approaches.

Accordingly, a simple QM family has been proposed, starting from a very general model of Web site, mapping its main logical components to the actors responsible for their quality. This QM defines the characteristics down to the second level: it is general enough to be applicable to a very large class of sites and to be used as a viable table of contents for requirement definition documents. It should be specialized and tailored for specific classes of Web sites and applications, intended purposes and

[8] According to ISO/IEC 25010, "any quality requirement, quality specification, or evaluation of quality that conforms to this International Standard shall either; a)- use the quality models defined in it or b)- tailor the quality model giving the rationale for any changes and provide a mapping between the tailored model and the standard model".

organizations, typically by dropping the sub-characteristics which are not relevant to the particular context, and defining lower levels of the hierarchy.

A comparison with the ISO QMs for software and software intensive systems has shown differences and similarities, originating from the particular nature of Web sites and applications, and the approach adopted in the QM construction. The proposed QM is essentially a superset and an abstraction of the ISO/IEC 25010 Product QM, where the common parts are allocated differently in the hierarchy of characteristics, mainly to comply to the organization mapping requirement.

7 References

1. ISO/IEC 25000:2005: Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE (2005)
2. Polillo, R.: Plasmare il Web – Road map per siti di qualità. Apogeo, Milano (2006)
3. Mich, L, Franch, M., Gaio, L.: Evaluating and Designing the Quality of Web Sites. Journal IEEE Multimedia, Vol.10, Issue 1, 34--43 (2003)
4. Polillo, R.: Il Check-up dei Siti Web. Apogeo, Milano (2004)
5. Signore, O.: A Comprehensive Model for Web Sites Quality. In: Proc. of WSE 2005 – 7th IEEE Int. Symposium on Web Site Evolution – Budapest, pp. 30--36 (2005)
6. Moraga, A., Calero, C., Piattini, M.: Comparing Different Quality Models for Portals. Online Information Review, Vol.30, Issue 5, pp. 555--468 (2006)
7. Malak, G., Sahraoui H., Badri, L., Badri, M.: Modeling Web Quality Using a Probabilistic Approach: An Empirical Evaluation. ACM Transactions on the Web, Vol.4, n.3 (2010)
8. Stefani, A., Xenos, M.: E-Commerce System Quality Assessment Using a Model Based on ISO 9126 and Belief Networks. Software Quality Control, Vol.16, Issue 1, pp. 107--129 (2008)
9. Herrera, M., Moraga, M.A., Caballero, I., Calero C.: Quality in Use Model for Web Portals (QiUWeP). In: ICWE 2010 – Proc. of the 10th Int. Conf. on Current Trends in Web Engineering. Springer, Heidelberg. pp. 91--101 (2010)
10. Lew, P., Olsina, L., Zhang, L.: Quality, Quality in Use, Actual Usability and User Experience as Key Drivers for Web Application Evaluation. In: Benatallah, B., Casati, F., Kappel, G., Rossi, G. (eds) ICWE 2010. LCNS, vol. 6189, pp. 218--232. Springer, Heidelberg (2010)
11. Moraga, C., Moraga, M.A., Calero, C., Caro, A.: SQuaRE-Aligned Data Quality Model for Web Portals. In: QSIC 2009 Proc. of the 2009 9th Int. Conf. on Quality Software. IEEE Press. pp. 117--122 (2009)
12. ISO/IEC 9126:1991: Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for their Use (1991)
13. ISO/IEC 9126-1:2001: Software Engineering – Product Quality – Part 1: Quality Model (2001)
14. ISO/IEC 25010:2011: System and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – System and Software Quality Models (2011)
15. ISO/IEC 9241-11:1998: Ergonomic Requirements for Office Work with Visual Display Terminals – Part 11: Guidance on Usability (1998)
16. ISO/IEC 25012:2008: Software Engineering – Software Product Quality Requirements and Evaluation (SQuaRE) – Data Quality Model (2008)
17. Morville, P., Rosenfeld, L.: Information Architecture for the World Wide Web. Third Edition. O'Reilly Media (2007)
18. Farmer, F.R., Glass, B.: Building Web Reputation Systems. O'Reilly Media (2010)