Software Qual J (2007) 15:155–177
DOI 10.1007/s11219-006-9008-x

Improving a portlet usability model

Mª Ángeles Moraga · Coral Calero · Mario Piattini · Oscar Diaz

Published online: 23 January 2007

Abstract Second-generation portals are far from being monolithic pieces of software. Their complexity calls for a component-based approach where portlets are the technical enabler. That being the case nowadays portals tend to be constructed by means of portlets, i.e. a multi-step, user-facing application to be delivered through a Web application. The proposal for and ample support given to the WSRP (Web Services for Remote Portlets) portlet standard predict an emerging portlet market. A main requirement for the blossoming of this market is the existence of portlet quality models that assist portal developers to select the appropriate portlet. This paper focuses on usability. The aim, therefore, is to develop a usability model for portlets. The paper presents such a model and its realisation for a sample case.

Keywords Web portal · Portlet · Quality · Usability · Model

1 Introduction

Web Portals are emerging Internet-based applications enabling access to different sources through a single interface (Mahdavi et al., 2004) which provides personalization,

Mª Á. Moraga · C. Calero · M. Piattini
Alarcos Research Group, Department of Information System and Technology, University of Castilla-La Mancha, Spain

C. Calero
e-mail: coral.calero@uclm.es

M. Piattini
e-mail: mario.piattini@uclm.es

O. Diaz
ONEKIN Research Group, Department of Computer Science, University of the Basque Country, Spain
e-mail: oscar.diaz@ehu.es

Mª Á. Moraga (✉)
Escuela Superior de Informática, Universidad de Castilla-La Mancha, Paseo de la Universidad 4, Ciudad Real, 13071, Spain
e-mail: MariaAngeles.moraga@uclm.es

single sign on, content aggregation from different sources and hosts the presentation layer of Information Systems (Java Community Process, 2003). Originally coined to describe Web-based applications that provide organized access to the resources of the Internet through search engines (such as Yahoo, Google), the term "portal" has been applied to systems that differ widely in capabilities and complexity—from static Web pages providing links to resources on a given topic, to inter-organizational systems providing access to multiple heterogeneous data sources and applications (Smith, 2004).

The primary objective for a portal is to create a working environment users can easily navigate through (Collins, 2001). This implies that services/content for different departments within the organisation or even from third parties, need to be integrated within the portal realm. From this viewpoint, the portal acts as a container from other applications.

Akin with this view, portals have evolved from monolithic systems to loosely-coupled compounds that glue together distinct applications. The technical enabler is the **portlet**. A portlet can be thought as a Web component that comprises a full-fledged Web application to be delivered through the portal. The recent standard WSRP (Web Services for Remote Portlets) accounts for portlet interoperability so that for example, a portlet built with Oracle Portal can be delivered in IBM WebSphere and vice versa.

This scenario provides the technological infrastructure to make feasible a portlet market *à la COST* so that portals can deliver portlets being provided by third parties. Indeed, the Open Source Portlet Repository Project has been recently launched (Blattman et al., 2006) to foster the free and open exchange of portlets. The Portlet Repository is *"a library of ready-to-run applications that you can download and deploy directly into your portal with, in most cases, no additional setups or configurations"*. Other similar portlet-sharing sites include *Portlet Swap* (jboss.org) and *Portlet Exchange* (portletexchange.com).

However, the lack of a measure framework for selecting the most appropriate portlets impedes a systematic comparison among portlets providing a similar functionality. This situation can be overcome if appropriate quality models are in place. These models will assist portal administrators in ascertaining which portlet provider best fits the portal's needs. Furthermore, it is most important to notice that, quality assurance is a must for ensuring returning users (Offutt, 2002).

Therefore, it becomes necessary to define a quality model for each quality characteristic that affects that same aspect of the portlets. In the software quality model proposed in ISO/IEC 9126 (ISO/IEC, 2001), quality characteristics are: functionality, reliability, usability, efficiency, and portability. In this paper, we will mainly focus on usability. From a portlet perspective and basing on ISO/IEC 9126 (ISO/IEC, 2001), this characteristic refers to the capability of the portlet to be understood, learned or used under specified conditions. In particular, the portlet usability model proposed by Diaz et al. (2004) is enhanced. To this end, several attributes are defined together with different measures and usability levels. Finally this model is applied to a specific portlet.

The model is defined along the WSRP standard. We do not claim the model to be exhaustive, rather a starting point where new attributes and measures can be introduced as further experience is developed. Perhaps, in this case, the major problem would be to reassess the usability level of a portlet taking into account the new measures.

This paper is structured as follows: In Section 2 we give a brief overview of portlets, while in Section 3 the portlet usability model is shown. Section 4 shows the assessment of the portlet usability level, with Section 5 applying this model to a specific portlet. The last section summarizes this paper and proposes future work.

2 Introduction to portlets

A portlet is a multi-step, user-facing application to be delivered through a Web application (e.g. a portal). Portals aggregate one or more portlets into web pages, which are usually personalized or customized for individual users or groups of users (Linwood and Minter, 2004).

Portlets generate fragments. A fragment is a piece of markup (e.g. HTML, XHTML, WML) adhering to certain rules and it can be aggregated with other fragments to form a complete document (Java Community Process, 2003). In order to form a portal page, the fragment generated by a portlet is aggregated to the fragments generated by other portlets. The necessary steps in the creation of a portal page include:

- The portlet container receives the content generated by the portlets and hands the portlet content to a portal.
- The portal server adds a title, control buttons and other decorations to the fragment generated by the portlet. As a result, the portal generates a portlet window.
- The portal server creates the portal page with the portlet windows.
- The portal server sends the portal page to the client device where it is displayed to the user.

Portlets have modes. A mode is a way of behaving. Five modes are distinguished, namely (OASIS, 2003): *view* (portlet should render markup reflecting the current state of the portlet), *edit* (portlet should provide content and logic that let a user customize the behaviour of the portlet), *help* (portlet may provide help screens that explain the portlet and its expected usage), *preview* (portlet should provide a visual sample of how this portlet will appear on the End-User's page with the current configuration) and *custom* (to declare additional custom modes).

Portlets cannot be deployed alone. A portlet is to be included within a third application (e.g. a portal framework). This can imply the portlet to be tuned to account for the idiosyncrasies of the hosting portal. Furthermore, the likely rendering of various portlet, simultaneously, makes the functionality of the portlet to be more focused than a traditional, stand-alone Web application.

2.1 Portlet standards

When portlets came into existence, each portal IDE (Independent Development Environment) vendor had a different API (Application Program Interface) for developing portlets, therefore portlets presented a lack of interoperability. However, the delivery of the Web Services for Remote Portlets (WSRP) specification overcomes this problem.

The goal of the WSRP standard is to enable an application designer or administrator to pick from a rich choice of compliant remote content and application providers, and integrate them with just a few mouse clicks and no programming effort (OASIS, 2003). To do this, WSRP defines four interfaces (OASIS, 2005):

- *Service Description Interface*: a required interface that allows consumers to find out the capabilities of the producer and about the portlets it hosts, including the metadata necessary for a consumer to interact with each portlet properly.
- *Markup Interface*: a required interface, used to request the generation of markup and the processing of interactions with that markup.

– *Registration Interface*: an optional interface, used to establish a relationship between a producer and a consumer.
– *Portlet Management Interface*: the purpose of this optional interface is to let consumers manage the persistent state and lifecycle of portlets explicitly.

The WSRP standard also sets out other characteristics that will be used to define the several attributes that affect the usability characteristic. For example, the WSRP standard defines modes and window states. Depending on the portlet mode, the portlet renders different content and performs different activities. Depending on the portlet window state, the portlet generates different lengths/styles of markup. The amount of page space that is assigned to the content generated is also different.

2.2 Portlets vs. Web applications: A usability perspective

A Web application can be considered as a software system that exploits the WWW infrastructure to offer its users the opportunity to modify the status of the system and of the business it supports (Conallen, 2003). In this sense, a portlet is a Web application. However, the component and reusable nature of portlets make them exhibit some specificities. In particular, differences between Web applications and portlets basically stem from the distinct running setting and the targeted end-users of each type of software (Diaz and Paz, 2005).

From a usability viewpoint, Web applications and portlets differ in:

- a Web application is self-sufficient whereas a portlet needs to be included into a hosting portal. This implies that usability should look not only at the end-user but also to the portal administrator.
- Web applications hardly support modes, whereas this is an essential part of a portlet. Some of these modes are used to facilitate portlet understanding and operational mode. These are key issues from a usability perspective.
- portlets need to support personalisation mechanisms to a larger extent than the typical Web applications. The portlet needs to be customised to the aesthetic guidelines of the hosting portal. Moreover, the most popular containers for portlets, portals, have personalization as one of their hallmarks. A portal should tune services and contents based on the current user profile. And this percolates the portal's components, i.e. the portlets.

Regardless of whether portlets and Web applications have some different characteristics, usability is a quality that affects both of them.

To date, several studies related to the usability of Web applications have appeared. For example, (Fraternali et al., 2004) present an evaluation framework that takes advantage of a conceptual modelling method for supporting the usability analysis of Web applications; (Ricca, 2004) has investigated, defined and applied a variety of conceptual tools, analysis, testing and re-structuring techniques able to support the quality of Web applications with the aim of assessing the quality of Web applications during their development and evolution; (Offutt, 2002) identified reliability, usability and security as the three most important quality criteria for Web application success; (Constantine and Lockwood, 2005) affirm that usability and user experience are emerging as critical determinants of success in Web applications because if consumers cannot find what they are looking for, they cannot buy it.

Despite the publication of many works focus on usability of Web applications none of them is enough for being directly applied to the portlet context due to the established differences among them. Therefore, our aim is to develop a usability model for the specificities of

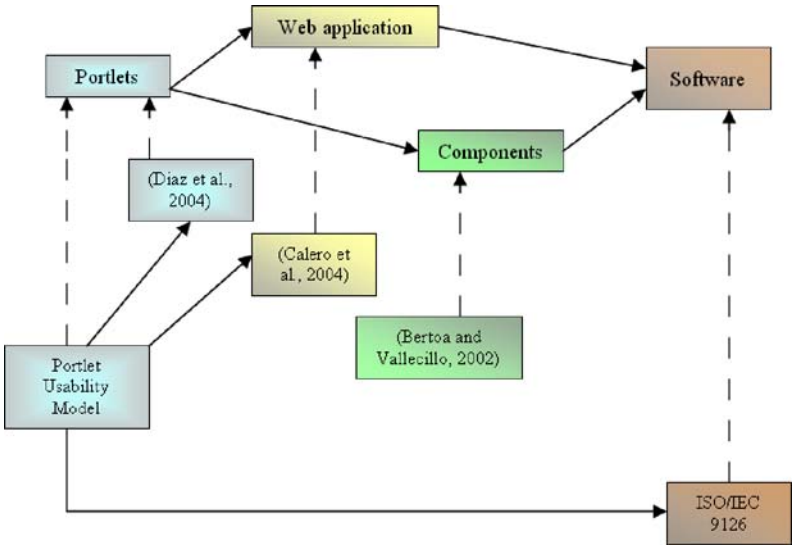

Fig. 1 Portlet usability model major influences

portlets. This model could be used to choose the most usable portlet among a set of portlets with similar functionality.

3 Portlet usability model

Despite its importance in being the only mechanism that makes users come back to the portal (Offutt, 2002), portal quality has scarcely been studied. Nowadays, portals tend to be made up of aggregated portlets. Therefore, in order to assure portal quality, firstly, it is necessary to improve the quality of the portlets as the atoms that conform the portal. This paper focuses on single portlets and it does not address portlet composition.[1]

There exist different aspects that affect portlet quality. Portlet reusability is addressed in Moraga et al. (2005). This paper focuses on usability as one of the most influential factor for both business success and user's satisfaction in a Web setting (Zhang, 2003). Therefore we think that the same study is essential for portlets.

Portlets sit in between Web applications and Software Components. Hence our usability model is elaborated from three main sources: the ISO/IEC 9126 standard, quality models for components, and quality models for Web applications. Figure 1 outlines the main references that influence this work. In this figure the scope of the models is denoted via the dotted lines and the influences are denoted via continuous lines.

The general quality framework is that of (ISO/IEC, 2001). To elaborate on the component side, the model proposed by Bertoa and Vallecillo (2002) has been mainly considered due to its completeness. To consider the Web characteristics, the main influence come from the

[1] However, the quality models developed for a single portlet could be applied in order to assess the quality of a portlet which is composed of other portlets (to do this, once the quality of the different portlets is assessed, the quality of portlets interactions should be analysed).

Table 1 Dimensions proposed for usability

Characteristic	Dimension	Definition	Relevance
Usability	Understandability	Capability of the portlet to enable the user to understand what the portlet is about	The portal administrator needs to understand the portlet functionality in order to integrate it into the portal
	Learnability	Capability of the portlet to enable the user to learn how the portlet achieves its aim	The portlet must be easy to learn
	Customizability	It refers to the attributes of portlet that enable the portlet to be customized by the user, to reduce the effort required to use it and also to increase satisfaction with the portlet	Users must be able to customize the portlet
	Compliance	Capability of the portlet to adhere to standards, conventions or regulations in laws and similar prescriptions relating to usability.	We must take into account the set of standards that the portlet adheres

Web quality model developed by Calero et al. (2004). Finally, this work rests on some early attempt described at Diaz et al. (2004).

In Table 1, the definition of the different dimensions along with their relevance is shown.

Moreover, once that the dimensions have been identified, attributes and measures have been defined. Specifically, the ones proposed by Díaz et al. (2004) have been considered as a starting point. However, some attributes and its associated measures have been updated. Specifically:

- For the understandability dimension:
  - The attribute "documentation/user manuals" is replaced by "documentation" due to the fact that portlets are able to provide online, hypertext documentation which is easier to access and browse than traditional documentation. Notice that the difference stems not from the supporting media (paper vs. digital) but the hypertext facilities built-in.
  - The attribute "help mode" is eliminated because the objective of this mode is to explain the portlet, therefore it does not provide additional information in order to use the portlet.
  - The attribute "documentation language" is added, because it is not only important that the portlet provides documentation about itself but also that this documentation can be read by the majority of potential users.
  - The attribute "description" is introduced since portlet understanding is enhanced if a brief summary about its functionality is provided.

- For the learnability dimension:
  - The attribute "documentation/user manuals" is replaced by "documentation" for the same reason that in the previous dimension.
  - The attributes "On line help" and "help mode" are replaced by "help". This attribute includes the other two attributes.

- The attribute "preview mode" is eliminated because this mode only provides a visual sample of how the portlet will appear on the portal with the current configuration. Therefore, it does not make easier to learn how the portlet achieves its aim.
- The attribute "vendor support" is eliminated because portlet learnability is not better if the portlet vendor gives support to the user.
- The attribute "screens" is introduced since the more screens the portlet needs to achieve one functionality, the more difficult is its learnability.
- The attribute "structured presentation" is added because a friendly presentation makes easier portlet learnability.

• For the customizability dimension:

- The attributes "necessary parameters", "categories of users", and "content depends on configuration" are added because all of them take into account portlet characteristics that facilitates its personalization.

• For the compliance dimension:

- The attributes "interface standards", "extensions" and "implementation standards" are eliminated since we focus on WSRP-compliant portlets and then, the support of WSRP is no longer a discriminant aspect between portlet providers.
- The attribute "standard compliance" has been added in order to consider if the portlet adheres to usability standards.

We only consider WSRP-compliant portlets (although the model can also be applied to a portlet that does not adhere to this standard, as it will be shown in the example). This standard defines interfaces, hence, the only attributes definable are those that can be calculated using information obtained from their interfaces or information provided by the vendors. Therefore, all the measures are objective and in consequence, the portlet usability model is generic and objective.

It can be worth emphasizing that the end-user of this model will be a portal developer who will want to choose the more usable portlet among a set of portlets with similar functionality. For this reason, the model assesses the usability of a portlet but it does not assess the usability perceived by the person who will use the portlet when integrated into a portal.

3.1 Understandability dimension

Based on (ISO/IEC, 2001), understandability is defined as the capability of the portlet to enable the user to understand what the portlet is about. Therefore, it bears on the users' effort for recognizing the logical concept and its applicability.

Following (ISO/IEC, 2001), understandability measures should be capable of evaluating the behaviour of users who have no previous knowledge on portlet operation. It should also be able to measure their difficulty in understanding portlet functions, operation and concepts.

The attributes and the measures set out for understandability are shown in Table 2.

The definition of the understandability attributes can be found in Appendix A.1.

3.2 Learnability dimension

Learnability is the capability of the portlet to enable the user to learn how the portlet achieves its aim.

Table 2 Attributes and measures proposed for the understandability dimension

Dimension	Attribute	Measure	Measure domain
Understandability	Interface language	Number of languages supported by the Interface	Natural number
	Documentation	The portlet vendor provides documentation	Boolean (True/False)
	Documentation language	Number of languages in which the documentation is written	Natural number
	Description	The portlet specifies its functionality	Boolean (True/False)

Table 3 Attributes and measures proposed for the learnability dimension

Dimension	Attribute	Measure	Measure domain
Learnability	Help	The portlet provides help	Natural number (Among 0 and 2)
	Documentation	The portlet vendor provides documentation	Boolean (True/False)
	Predictability	Portlet predictability	List (1...5): very difficult, difficult, normal, easy, very easy
	Screens	Number of screens in order to achieve one functionality	Natural number
	Structured presentation	Structured presentation degree	List (1...5): very difficult, difficult, normal, easy, very easy

Following (ISO/IEC, 2001), learnability measures should be capable of evaluating or drawing the user curve of performance on portlet operation, from a starting point of no knowledge about the evaluated portlet. An external learnability measure should be able to measure such an attribute as the behaviour of a user who is learning how to use the portlet.

In Table 3, we set out the attributes and the measures that affect learnability.

In Appendix A.2, the definition of the previous attributes is detailed.

3.3 Customizability dimension

Customizability refers to the attributes of software that enable the software to be customized by the user, to reduce the effort required to use it and also to increase satisfaction with the software.

In Table 4, the different attributes and measures that affect portlet customizability are shown.

The definition of the attributes can be found in Appendix A.3.

3.4 Compliance dimension

Compliance is the capability of the portlet to adhere to standards, conventions or regulations in laws and similar prescriptions relating to usability.

Following (ISO/IEC, 2001), a usability compliance measure should be able to measure an attribute such as the number of functions with, or occurrences of, compliance problems, consisting in the portlet failing to adhere to required standards, conventions, style guides or regulations relating to usability.

Table 4 Attributes and measures proposed for customizability dimension

Dimension	Attribute		Measure	Measure domain
Customizability	Location		Location availability	Boolean (True/False)
	Localization	Languages	Number of languages	Standard language list
		Country	Number of countries	Standard country list
	Time		Time adaptation availability	Boolean (True/False)
	Device	Markup type	Markup types that the portlet supports	List
		User agent	User agents that the portlet supports	List
	Network		Network adaptation availability	Boolean (True/False)
	User		Number of user profile characteristics that the portlet stores	Natural number
	Window State		Number of additional window states supported by the portlet	Natural number
	CSS		CSS availability	Boolean (True/False)
	Edit mode		Edit mode availability	Boolean (True/False)
	Necessary parameters		Ratio of necessary parameters	Float number
	Categories of users		Number of categories of users	Natural number
	Content depends on configuration		Content generated by the portlet depends on the configuration	Boolean (True/False)

Table 5 Attributes and measures proposed for compliance dimension

Dimension	Attribute	Measure	Measure domain
Compliance	Standard compliance	The portlet adheres to usability standards	Natural number

The attribute for compliance dimension is shown in Table 5.
This attribute is defined in Appendix A.4.

4 Level of usability

Once attributes and measures have been set, the portlet usability level must be ascertained. To do this, we first need to assess the portlet understandability, learnability, customizability and compliance levels, independently. We wish to emphasize that the measures values which have been established for each level, are WSRP-compliant.

Moreover, considering the set of values that the different measures can acquire, and with the aim of defining a function that assesses the portlet level for the usability dimensions, we have established that all the measures can acquire a value among 0 and 5. This goes along the following lines:

1. measure values are mapped into a ranking from 0 to 5.
2. levels are established (e.g. excellent, high, middle, acceptable and non-acceptable) and a function is defined to assess the quality level of the different dimensions.

This function takes the form:

$$F(\text{Dimension}_x) = \textit{Adding\_Ranking\_Measures}$$

where: Dimension_x = represents one of the usability dimensions (i.e. understandability, learnability, customizability and compliance).

Adding_Ranking_Measures = stands for the addition of the values of the ranking of the different measures defined for the Dimension_x.

In fact, the function aim is twofold. First, the function is used to establish the different levels for each usability dimension (i.e. excellent, high, middle, etc.). This in turn, implies (1) setting both the maximum and minimum values of the function and (2) the values in between. The second aim is to assess the usability level of a concrete portlet.

4.1 Portlet understandability, learnability, customizability and compliance level

4.1.1 Step 1: Ranking mapping

Firstly, measure values are mapped into a ranking from 0 to 5. This mapping can be adapted in accordance with the measure itself, the portal developer's need or the type of portal that will deliver the portlet. After all, this mapping attempts to capture the subjectivity that goes with quantifying qualitative notions. As an example, the importance of the measure *"number of languages in which the documentation is written"* very much depends on the potential market and end-user sensibility on the language issue. For instance, European portals tend to live in a more multi-lingual setting which can lead that the maximum punctuation will require support for a larger number of languages that would be required in a mono-lingual setting. In this way, an adaptative mapping offers a gateway to tune the quality model to "the user subjectivity model"

As a matter of example, in the following paragraphs we will show how to define the usability level of a portlet. We wish to emphasize that the presented values can be changed according to the needs of the portal in which the portlet will be aggregated, making the model adaptable to each user's preferences. Tables 6–8 provide an example for understandability measures. In addition, both the minimum (i.e. lowest acceptable value) and maximum values are set.

In Tables 9–11 we show an example of how the measure values of the learnability dimension can be mapped into the ranking.

The ranking for the measure "the portlet vendor provides documentation" is given in Table 7, whereas Tables 12–16 provide the counterparts for measures of customizability dimension.

The rankings for the measures "location availability", "CSS availability" and "edit mode availability" of the customizability dimension are the same as Table 7, whereas Table 17 gives the mapping for the measure of compliance dimension.

4.1.2 Step 2: Level establishment

In the second step the different dimensions levels must be assessed, i.e. understandability, learnability, customizability and compliance levels. To this end, the maximum and the

Table 6 Ranking for "number of languages supported by the Interface" measure

Ranking	Measure value
0	1
1	1
2	1
3	1
4	2
5	>2

$\frac{0+1+2+3}{4} = 1.5$ =>Non-accep

4=>Minimum acceptable Value (V_{min})

5=> Maximum Value (V_{max})

Table 7 Ranking for "the portlet vendor provides documentation" and "the portlet specifies its functionality" measures

Ranking	Measure value
0	0
1	0
2	0
3	1
4	1
5	1

1=>Non-accep

4=>V_{min}=V_{max}

Table 8 Ranking for "number of languages in which the documentation is written" measure

Ranking	Measure value
0	1
1	1
2	2
3	2
4	>2
5	>2

0.5=>Non-accep

2.5=>V_{min}

4.5=>V_{max}

minimum values of the function F for each dimension must be assessed. These values must be defined by the portal developer taking into account his needs, the kind of portal in which the portlet will be aggregated and so on.

Going on with our example, we consider that the function acquires the minimum (or maximum) acceptable value when all the measures of this dimension have obtained the minimum (or maximum) value. These values are shown in Table 18.

Table 9 Ranking for "the portlet provides help"[a]

Ranking	Measure value	
0	0	0.5=>Non-accep
1	0	
2	1	2.5=>V_{min}
3	1	
4	2	4.5=>V_{max}
5	2	

[a]The maximum value for this measure is 2 which corresponds to the two different ways in which help can be delivered, namely, the *view* mode and *help* mode. The former is the common on-line help for guiding the user while achieving the main functionality of the portlet (e.g. buying a train ticket). By contrast, the *help* mode is decoupled from this main functionality, and can be enacted at any time, providing help screens that explains the portlet and its expected usage. Therefore, the value of this measure can vary between 0 and 2.

Table 10 Ranking for "portlet predictability" and "structured presentation degree"

Ranking	Measure value	
0	Very difficult/1	0.5=>Non-accep
1	Very difficult/1	
2	Difficult/2	2=>V_{min}
3	Normal/3	
4	Easy/4	
5	Very easy/5	5=>V_{max}

Table 11 Ranking for "number of screens in order to achieve one functionality"

Ranking	Measure value	
0	>5	0=>Non-accep
1	5	1=>V_{min}
2	4	
3	3	
4	2	
5	1	5=>V_{max}

Table 12 Ranking for "number of languages", "number of countries", "markup types that the portlet supports" and "user agents that the portlet supports"

Ranking	Measure value	
0	0	0=>Non-accep
1	1	
2	1	1.5=>Vmin
3	2	
4	2	
5	≥3	5=>Vmax

Table 13 Ranking for "time adaptation availability", "network adaptation availability", "number of categories of users" and "content generated by the portlet depends on the configuration"

Ranking	Measure value	
0	0	
1	0	1=>Vmin
2	0	
3	1	
4	1	4=>Vmax
5	1	

Table 14 Ranking for "number of user profile characteristics that the portlet stores"[a]

Ranking	Measure value	
0	CU < 11	0=>Non-accep
1	11≤ CU < 14	1=>Vmin
2	14≤ CU < 17	
3	17≤ CU < 20	
4	20≤ CU < 23	
5	CU ≥ 23	5=>Vmax

[a]The WSRP standard identifies 69 different characteristics about the end-user that a portlet can store. In the example, we consider the storage of a third of them to be a sign of outstanding concern for the "number of user profile characteristics that the portlet stores", whereas the storage of only 11 (a sixth of the total number) would be non-acceptable.

After the minimum and maximum acceptable values have been assessed, these values are divided in order to obtain the intermediate levels. In our example, equitable intervals have been established although this is up to the portal developer Tables 19–22 show the understandability, learnability, customizability and compliance levels, respectively.

It should be noted that the non-acceptable level has not been identified for the compliance dimension. This stems from the fact that the only measure defined for this dimension (i.e.

Table 15 Ranking for "number of additional window states supported by the portlet"[a]

Ranking	Measure value	
0	0	0.5=>Non-accep
1	0	
2	1	2=>Vmin
3	2	
4	3	
5	4	5=>Vmax

[a]The measure value can vary between 0 and 4 due to WSRP standard defines five different window state but one of them is the *normal window state* which is compulsory. Hence, the number of additional window states is limited to four: *minimized, maximized, solo and custom*. The rationales for these *window states* can be found in page 27.

Table 16 Ranking for "ratio of necessary parameters"

Ranking	Measure value	
0	2 < RP	0=>Non-accep
1	1.75<RP ≤ 2	1=>Vmin
2	1.5<RP≤ 1.75	
3	1.25<RP ≤ l.5	
4	1 < RP ≤ 1.25	
5	RP = 1	5=>Vmax

Table 17 Ranking for "the portlet adheres to usability standards"

Ranking	Measure value	
0	0	0.5=>Vmin
1	0	
2	1	
3	2	
4	3	
5	≥ 4	5=>Vmax

Table 18 Minimum and maximum values of the function F for the different dimensions

Understandability	Learnability	Customizability	Compliance
$F(\text{Understandability})_{min} = 14.5$	$F(\text{Learnability})_{min} = 11.5$	$F(\text{Customizability})_{min} = 26$	$F(\text{Compliance})_{min} = 0.5$
$F(\text{Understandability})_{max} = 17.5$	$F(\text{Learnability})_{max} = 23.5$	$F(\text{Customizability})_{max} = 64$	$F(\text{Compliance})_{max} = 5$

Table 19 Understandability levels

Understandability level	Set of values
Excellent	F (Understandability) = 17.5
High	16.5 ≤ F (Understandability) < 17.5
Middle	15.5 ≤ F (Understandability) < 16.5
Acceptable	14.5 ≤ F (Understandability) < 15.5
Non-acceptable	F (Understandability) < 14.5

Table 20 Learnability levels

Learnability level	Set of values
Excellent	F (Learnability) = 23.5
High	19.5 ≤ F (Learnability) < 23.5
Middle	15.5 ≤ F (Learnability) < 19.5
Acceptable	11.5 ≤ F (Learnability) < 15.5
Non-acceptable	F (Learnability) < 11.5

Table 21 Customizability levels

Customizability level	Set of values
Excellent	F (Customizability) = 64
High	51.3 ≤ F (Customizability) < 64
Middle	38.7 ≤ F (Customizability) < 51.3
Acceptable	26 ≤ F (Customizability) < 38.7
Non-acceptable	F (Customizability) < 26

Table 22 Compliance levels

Compliance level	Set of values
Excellent	F (Compliance) = 5
High	3.5 ≤ F (Compliance) < 5
Middle	2 ≤ F (Compliance) < 3.5
Acceptable	0.5 ≤ F (Compliance) < 2

portlet adhering to usability standards) is not applicable as there is not such standards for portlets. Therefore, we have been less restrictive and a portlet compliance level can be acceptable despite not adhering to standards. Of course, this consideration is up to the portal developer.

4.2 Portlet usability level

4.2.1 Step 1: Ranking mapping

Once the portlet level for each dimension is set, the portlet usability level is ascertained as an aggregated of these levels. This implies assigning a value to each level (see Table 23).

4.2.2 Step 2: Level establishment

Next, we define a function in order to assess the usability level. The function value is obtained adding the levels to the different dimensions:

$$\begin{aligned} \text{F(Usability)} = {} & \text{Understandability\_Level} + \text{Learnability\_Level} + \text{Customizability\_Level} \\ & + \text{Compliance\_Level} \end{aligned}$$

Table 23 Ranking for dimension levels

Ranking	Dimension Level	
0	Non-Accept.	0=>Non-accep
1	Acceptable	1.5=>Vmin
2	Acceptable	
3	Middle	
4	High	
5	Excellent	5=>Vmax

Table 24 Usability levels

Usability	Set of values
Excellent	F (Usability) = 20
High	15.82 ≤ F (Usability) < 20
Middle	11.66 ≤ F (Usability) < 15.82
Acceptable	7.5 ≤ F (Usability) < 11.66
Non-acceptable	F (Usability) < 7.5

Next, the maximum and minimum values of the function are set:

$$\begin{aligned}
F(\text{Usability})_{\max} &= (\text{Understandability_Level})_{\max} + (\text{Learnability_Level})_{\max} \\
&\quad + (\text{Customizability_Level})_{\max} + (\text{Compliance_Level})_{\max} \\
&= \text{Excellent} + \text{Excellent} + \text{Excellent} + \text{Excellent} = 5+5+5+5=20 \\
F(\text{Usability})_{\min} &= (\text{Understandability_Level})_{\min} + (\text{Learnability_Level})_{\min} \\
&\quad + (\text{Customizability_Level})_{\min} + (\text{Compliance_Level})_{\min} \\
&= \text{Acceptable} + \text{Acceptable} + \text{Acceptable} + \text{Acceptable} \\
&= 1.5+1.5+1.5+1.5=6
\end{aligned}$$

Once the minimum and maximum acceptable values are set, intermediate levels are obtained. In our case, the intermediate levels are obtained as equitably divisions. The different usability levels are summarized in Table 24.

5 Applying the usability model

A portlet is used as a testbed for this usability model. The portlet permits staff to consult salary-related information. The portlet has been developed by OCU, S.A. (Oficina de Cooperacion Universitaria) a company created in 1994 by seven Spanish Universities and the Santander Bank. OCU develops and implements different software applications for academic management in Spain and Latin America.

Dimension	*Measure*	*Value*
UNDERSTAN-DABILITY	Number of languages supported by the Interface	+100
	The portlet vendor provides documentation	1
	Number of languages in which the documentation is written	1
	The portlet specifies its functionality	1

⇨ Excellent understandability level

Fig. 2 Results for understandability

Dimension	*Measure*	*Value*
LEARNABILITY	The portlet provides help	1
	The portlet vendor provides documentation	1
	Portlet predictability	Very easy
	Number of screens in order to achieve one functionality	1
	Structured presentation degree	Easy

⇨ High learnability level

Fig. 3 Results for learnability

The portlet obtains the user's NIF by means of the information that the user provides when logs in. Next, the portlet gets information about the available payslips for this NIF and about what is the last one. All this information is obtained from a database and is rendered to the portlet user.

In the following figures, the portlet level for the different usability dimensions and the usability level are shown. Figure 2 shows the results for the understandability dimension. Following our model, and according to the values shown in Fig. 2, the portlet understandability is "excellent" (i.e. F (Understandability) = 17.5).

In Fig. 3 the values for the learnability measures are presented As a result, the portlet learnability level has obtained the "high" value (F (Learnability) = 20.5).

The values obtained for the portlet customizability are shown in Fig. 4. Considering the defined levels for customizability in Fig. 4, the customizability level of this portlet is "middle" (F(Customizability) = 43.5).

Figure 5 shows the results for the compliance dimension which results in an "acceptable" value (F(Compliance) = 0.5) for our sample portlet.

Finally, these results are summed up in the usability characteristic whose outcome is "acceptable" (i.e. F(Usability) = 13.5) (Fig. 6).

It may be worth underlining that this portlet does not adhere to the WSRP standard which makes the portlet usability level to be low.

Having assessed the portlet usability level, we are in a position to provide some recommendations to improve this outcome. Specifically, the actions are:

(A) Actions to obtain an excellent learnability level. These include:

- The portlet should provide the "help" mode.
- The structure degree of the presentation should be "very easy".

(B) Actions to improve the customizability level. These include:

- The portlet should be able to adapt to the network.

Dimension	*Measure*	*Value*
CUSTOMIZABILITY	Location availability	1
	Number of languages	+100
	Number of countries	+100
	Time adaptation availability	1
	Markup types that the portlet supports.	2
	User agents that the portlet supports.	4
	Network adaptation availability	0
	Number of user profile characteristics that the portlet stores.	1
	Number of additional window states supported by the portlet	0
	CSS availability	1
	Edit mode availability	0
	Ratio of necessary parameters	1
	Number of categories of users	1
	Content generated by the portlet depends on the configuration	0

⇨ Middle customizability level

Fig. 4 Results for customizability

Dimension	*Measure*	*Value*
COMPLIANCE	The portlet adheres to usability standards	0

⇨ Acceptable compliance level

Fig. 5 Results for compliance

Characteristic	*Dimension*	*Level*
USABILITY	Understandability	Excellent
	Learnability	High
	Customizability	Middle
	Compliance	Acceptable

⇨ Middle usability level

Fig. 6 Results for usability

- The portlet should store at least 23 user profile characteristics, due to this is the minimum acceptable value.
- The portlet should support four additional window states because this is the number of window states that WSRP standard identifies.
- The portlet has to support the "edit" mode.
- The content generated by the portlet should depend on its configuration.

Finally, the action for obtaining a better compliance level is to ensure that the portlet adheres to one or more usability standards.

6 Conclusions and future work

The WSRP standard makes possible the existence of a portlet market *à la COST*. This market requires the existence of portlet quality models that assist portal developers in selecting the appropriate portlet.

This paper proposes a usability model for portlets where different attributes have been identified that affect each usability dimension. Moreover, we have defined measures for each of these attributes. As a first attempt, the model has been applied to a real portlet. The output serves not only to indicate the adequacy of the portlet but to guide the portlet developer in detecting the weaknesses of the portlet as far as usability is concerned.

In future work, we will have to work on the validation of the presented model, trying to assess if the usability model is complete or if on the contrary, some dimensions or measures must be eliminated, aggregated or redefined. We also plan to define a complete portlet quality model.

As a portal can be regarded as an aggregation of portlets, where portal quality could be derived from the quality of its constituents, our next step will be to assess the quality of a whole portal, taking into account the quality levels of the portlets used to construct it. Likewise, the identification of other factors that could affect the quality portal would be necessary.

Appendix A: Attributes definition

A.1 Understandability attributes definition

The meaning of the attributes is explained as follows:

- *Interface language*: WSRP portlets have a generic interface to achieve interoperability (similar to Servlets). That being so, the interface as such does not give us any hint about the aim of the portlet. Therefore we only consider as an attribute the interface language defined as: "the portlet interface supports different languages".
- *Documentation*: the portlet vendor provides the portlet with documentation on line. Hence additional information, which can help the portal administrator to understand the portlet, is provided.
- *Documentation language*: the documentation is provided in several languages.
- *Description*: this refers to the existence of a description of the portlet functionality, helping the end-user to understand it. This information can be obtained through the method *getPortletDescription*.

A.2 Learnability attributes definition

The meaning of the attributes established for this dimension is:

- *Help*: the portlet vendor provides help. Its domain is a natural number among 0 and 2, where 0 means that the portlet does not provide on-line help or help mode, 1 means that it provides one of them and 2 means that it provides both.
- *Documentation*: the portlet vendor provides the portlet with documentation on line. Therefore, additional information is provided.
- *Predictability*: portlet interface icons are easily related to the actions the portlet performs.

– *Screens*: it refers to the number of screens for achieving one functionality.
– *Structured presentation*: the presentation of the portlet is structured and easy to understand.

A.3 Customizability attributes definition

The meaning of the attributes is explained as follows:

– *Location*: the portlet captures information about the location from which it is accessed.
– *Localization*: is the capacity to tailor one portlet to the idiosyncrasies of a given culture-this is becoming an increasing concern. The aspects of cultural diversity that need specific support are normally arranged around two features, namely, *language* and *country*.
– *Time*: the portlet allows the adaptation of the application with respect to certain timing constraints.
– *Device*: this attribute discusses the demand of ubiquitous Web applications for any media, in terms of multi-channel delivery, and it provides basic information about the hardware and software capabilities of the device accessing the application. This feature can be split into *Markup_Type* (e.g. HTML, WML20, OR VoiceXML), and *User_Agent* (e.g. Netscape 702, mise60, or nokia7650).
– *Network*: the portlet can adapt itself to different networks. This attribute considers adaptation from the network viewpoint, and whether network context information, e.g. bandwidth or package losses, affects the application.
– *User*: the portlet takes into account the personal characteristics of the user. This attribute regards the need for personalization, i.e. how the user profile (e.g. demographic data, knowledge, skills and capabilities, interests and preferences, goals and plans) is considered by the application.
– *Window states*: space left for portlet rendering. WSRP defines five window states: *normal*, indicates the portlet is in all likelihood sharing the aggregated page with other portlets; *minimized*, the portlet should not render visible markup, but it is free to include non-visible data such as JavaScript or hidden forms; *maximized*, specifies that the portlet is probably the only one being rendered in the aggregated page, or that the portlet has more space compared to other portlets in the aggregated page; *solo*, denotes that the portlet is the only portlet being rendered in the aggregated page; *custom*, for consumers to declare additional custom window states.
– *CSS*: the portlet considers aesthetic guidelines for preserving the identity of the portal.
– *Edit mode*: the portlet provides the end-user with a mode for configuring the portlet. Within this mode, a portlet should provide content and logic that let a user customize the behaviour of the portlet.
– *Necessary parameters*: relation between the number of parameters which are requested of the end-user and the number of parameters that the portlet really uses in order to adapt the portlet to him/her. In order to measure the number of necessary parameters we define the next ratio:

$$\text{Ratio of necessary parameters} = \frac{Parameters\_user\_fill\_in}{Parameters\_portlet\_use}$$

Where: Parameters_user_fill_in: is the number of parameters that the user must provide to the portlet. Parameters_portlet_use: is the number of parameters that the portlet really uses.

The ideal case is when "necessary parameters" takes the value 1 because this means that the portlet uses all the parameters it asks the user for.
- *Categories of users*. The portlet supports communities: the content generated depends on the category of the user who is interacting with the portlet.
- *Content depends on configuration*: The portlet can tailor its generated content (in the mode view) to specific users depending on the configuration (window state, categories of users, user profile, user's preferences, etc.).

A.4 Compliance attributes definition

The possible standards for portlet usability are:

- ISO/IEC 9241 compliance (ISO/IEC, 1999): provides detailed guidance on the design of user interfaces (parts 12–17). ISO/IEC 9241 provides requirements and recommendations relating to the attributes of the hardware, software and environment that contribute to usability, and the ergonomic principles underlying them.
- ISO/IEC 14915 compliance (ISO/IEC, 2000): contains recommendations for multi-media interfaces. It provides recommendations for navigation structures and aids, media controls, general guidelines for media selection and combination, etc.
- IEC CDV TR 61997 compliance (IEC CDB TR 61997, 2000): contains recommendations for multi-media interfaces. To be specific, this technical report gives general principles and detailed design guidance for media selection, and for mechanical, graphical and auditory user interfaces.
- ISO DTS 16071 compliance (ISO/IEC, 2000): this technical specification provides guidelines and recommendations for the design of systems and software that will enable users with disabilities to have greater accessibility to computer systems (with or without assistive technology). These users include those with low vision, users with impaired hearing, or who are deaf, those with physical and cognitive impairments, and the elderly.
- Usability Engineering Process Model compliance (Granollers et al., 2003): the Usability Engineering Process Model integrates software engineering with a set of well-organized activities addressing usability, such as structured requirement analysis activities where usability is vital from the very beginning, or iterative evaluation activities of the usability goals.

Acknowledgments The authors would like to thank the anonymous reviewers for their invaluable feedback. This work is part of the CALIPO project (TIC 2003-07804-C05-03) and CALIPSO network (TIN2005-24055-E) supported by the Ministerio de Ciencia y Tecnologia and the DIMENSIONS project (PBC-05-012-1) supported by FEDER and la Consejería de Educación y Ciencia de la Junta de Comunidades de Castilla-La Mancha.

References

Bertoa, M.F., Vallecillo, A. 2002. Quality attributes for COTS components. In: 6th International Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE'2002), Málaga, pp. 54–66.

Blattman, J., Krishnan, N., Polla, D., Sum, M. 2006. Open-Source Portal Initiative at Sun, Part 2: Portlet Repository Available on: http://developers.sun.com/prodtech/portalserver/reference/techart/portlet-repository.html#2. Accessed: 26-10-2006.

Calero, C., Ruiz, J., Piattini, M. 2004. A web metrics survey using WQM. In: Fourth International Conference on Web Engineering, Munich, vol. 147–160.

Collins, H. 2001. Corporate Portals. New York: Amacom.

Conallen, J. 2003. Building Web Applications with UML, 2nd edn. Addison-Wesley.
Constantine, L., Lockwood, L. 2005. Usage-Centered Engineering for Web Applications. Available on: http://www.foruse.com/articles/webapplications.pdf. Accessed: 24-09-2005.
Diaz, O., Calero, C., Piattini, M., Irastorza, A. 2004. Portlet usability model. IBM Research Report. RA221(W0411-084). In: ICSOC 2004., pp. 11–15.
Diaz, O., Paz, I. 2005. Turning web applications into portlets: raising the issues. In: Symposium on Applications and the Internet (SAINT'05). IEEE Computer Society, pp. 31–37.
Fraternali, P., Lanzi, P.L., Matera, M., Maurino, A. 2004. Exploting conceptual modeling for web application quality evaluation. In: WWW 2004, pp. 342–343.
Granollers, T., Lorés, J., Perdrix, F. 2003. Usability engineering process model. Integration with software engineering. In: Procs. of HCI-Intl'03, Crete-Greece.
IEC CDB TR 61997. 2000. Guidelines for the user interfaces in multimedia equipment for general purpose use.
ISO/IEC. 2000. ISO/IEC DTS 16071: Guidance on accessibility for human-computer interfaces.
ISO/IEC. 2001. ISO/IEC 9126. Software Engineering-Product Quality. Parts 1 to 4. International Organization for Standardization/International Electrotechnical Commission.
ISO/IEC. 1999. ISO/IEC 9241. Ergonomic requirements for office work with visual display terminals.
ISO/IEC. 2000. ISO/IEC 14915. Software ergonomics for multimedia user interfaces.
ISO/IEC. 2001. ISO/IEC 9126-1 Software Engineering- Product Quality. Part 1: Quality Model.
Java Community Process. 2003. JSR 168 portlet specification. Available on: http://www.jcp.org/en/jsr/detail?id = 168. Accessed: January.
Linwood, J., Minter, D. 2004. Building Portals with the Java Portlet API. United States of America: Apress.
Mahdavi, M., Shepherd, J., Benatallah, B. 2004. A collaborative approach for caching dynamic data in portal applications. In: Proceedings of the Fifteenth Conference on Australian Database, vol. 27, pp. 181–188.
Moraga, M.Á., Calero, C., Paz, I., Diaz, O., Piattini, M. 2005. A reusability model for portlets. In: Web Information Systems Quality (WISQ 2005) Workshop, pp. 21–32.
OASIS. 2003. Web Service for Remote Portals (WSRP). Version 1.0. Disponible en: http://www.oasis-open.org/commitees/wsrp/. Accedido.
OASIS. 2005. Web Service for Remote Portals (WSRP) Version 2.0. Disponible en: http://www.oasis-open.org/commitees/wsrp/. Accedido: June.
Offutt, A.J. 2002. Quality attributes of web software applications. IEEE Software **19**(2):25–32.
Ricca, F. 2004. Analysis, testing and re-estructuring of web applications. In: 20th IEEE International Conference on Software Maintenance (ICSM'04), pp. 474–478.
Smith, M.A. 2004. Portals: toward an application framework for interoperability. Communications of the ACM **47**(10):93–97.
Washizaki, H., Yamamoto, H., Fukazawa, Y. 2004. A metrics suite for measuring reusability of software components. Software metrics symposium. In: Ninth International, pp. 211–223.
Zhang, X. 2003. Web usability study: a review. In: Ninth Americas Conference on Information Systems.

M[a] Ángeles Moraga received her M.Sc. in Computer Science and her Technical Degree in Computer Science by the University of Castilla-La Mancha (UCLM). Nowadays she is developing her Ph.D. at the UCLM. She holds a FPI grant (BES-2004-5206) from the Spanish Ministerio de Educación y Ciencia. She is a member of the Alarcos Research Group, in the same University, specialized in Information Systems, Databases and Software Engineering. Her research interests are: portals, software quality and measures.

Coral Calero is Ph.D. in Computer Science. Associate Professor at the Escuela Superior de Informática of the University of Castilla-La Mancha in Ciudad Real (Spain). She is a member of the Alarcos Research Group, in the same University. Her research interests are: database/data-warehouse quality, web/portal quality, software measures, empirical software engineering and ontologies.

Mario Piattini is Ph.D. in Computer Science. Full Professor at the Escuela Superior de Informática of the University of Castilla-La Mancha. Author of several books and papers on databases, software engineering and information systems. He leads the ALARCOS research group of the Department of Computer Science at the University of Castilla-La Mancha, in Ciudad Real, Spain. His research interests are: Information System quality, software metrics, software maintenance and security.

Oscar Díaz is full professor at the University of the Basque Country. He obtained the BSc in Computing at the University of the Basque Country, and a Ph.D. by the University of Aberdeen. Currently, he leads a fifteen-member group which focuses on R&D on Web Engineering. His current interests include portlet-based portals, software product lines and blogs in an academic context. He has over 50 international publications which include VLDB Journal, ACM TOIT; ACM Computing Surveys or IEEE Software.