

# A Reusability Model for Portlets\*

M<sup>a</sup> Ángeles Moraga<sup>1</sup>, Coral Calero<sup>1</sup>, Iñaki Paz<sup>2</sup>,  
Oscar Díaz<sup>2</sup>, and Mario Piattini<sup>1</sup>

<sup>1</sup> Alarcos Research Group, University of Castilla-La Mancha, Spain  
{MariaAngeles.Moraga, Coral.Calero, Mario.Piattini}@uclm.es

<sup>2</sup> ONEKIN Research Group, University of the Basque Country, Spain  
inaki.paz@ehu.es, oscar@si.ehu.es

**Abstract.** By means of portals, a company can give each person the information that responds to their specific needs. Nowadays, portals tend to be constructed by means of portlets. So, if we want “good” portals, we must select the most appropriate portlets for their construction. This requires the existence of appropriate quality models to assess portlet’s diverse characteristics. Perhaps one of the most important non-functional characteristics (from the point of view of the portlet consumer) is reusability. This work aims to define a reusability model for portlets that will allow the assessment of the portlet reusability level. As an example of application, we have applied the reusability model to a concrete portlet in order to know its level of reusability.

## 1 Introduction

In the last decade, portals have evolved from being simple providers of access web pages and corporate databases to become a support for intelligent management, application integration and collaborative processing (Collins, 1999). The advantage of portals is their ability to integrate and personalize several technologies (groupware, databases, data-warehouses, e-mail, meta-data, intelligent management systems, etc.) within a single business management tool (Kvitka, 2002). Aligned with this integration imperative, portlets have recently been proposed to integrate third-party applications into the portal realm. A portlet is a multi-step, user-facing application to be delivered through a Web application. So, if we want “good” portals, we must select the most appropriate portlets for constructing the portal. For this reason, the portlet market requires the existence of quality models. Quality, however, is a wide concept that must be defined according to other sub-characteristics. In the portlet context, there are two fundamental quality characteristics namely, usability, defined as the capability of the portlet to be understood, learned or used under specified conditions, and reusability, defined as the capability of the portlet to be used in different portals

---

\* This work was conducted when the first author was in stage at the University of the Basque Country and is part of 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. and the CALIPO project (TIC 2003-07804-C05-03) supported by the Ministerio de Ciencia y Tecnología.

by several developers. We have already defined a portlet usability model (Diaz et al., 2004). So this paper, focuses on the reusability characteristic.

This paper is structured as follows. Section 2 gives a brief on portlets. Section 3 explains the portlet reusability model. Section 4 shows the assessment of the portlet reusability level. Section 5 applies this model to a specific portlet. Last section summarizes this paper and proposes future work.

## 2 Portlets: A Brief Summary

A portlet is a multi-step, user-facing application to be delivered through a Web application (e.g. a portal). Its novelties could be best assessed by comparing portlets with a technology which is already well-known, such as Java Servlets. As with Servlets, Java portlets run in a portlet container, a server container that provides portlets with a running environment. Servlet generates HTML pages as a result of a browser invocation. Likewise, a portlet also generates XHTML fragments (or other markup language) to be framed by the invoker.

There are however, two main differences between these two technologies. Firstly, a Servlet is a one-step process, while a portlet comprises a multi-step process. A portlet comprises the state-handling and navigation logic that supports the multi-step process. Secondly, a Servlet can generate a full page. By contrast, a portlet generates fragments which are to be arranged along with fragments from other portlets to build up the final portal's page. Hence, a portal page can contain a number of portlets that users can arrange into columns and rows, minimizing, maximizing or customising to suit their individual needs. So far however, the lack of a common model prevents portlet interoperability. This hinders a portlet which has been developed in, let's say, Oracle Portal, being deployed at a Plumtree portal and vice-versa. However, the delivery of the Web Services for Remote Portlets (WSRP) specification overcomes this problem. WSRP defines four interfaces (OASIS, 2005), Service Description Interface, Markup Interface, Registration Interface and Portlet Management Interface, and also other characteristics that will be used to define the several attributes that affect the reusability characteristic. For example, the WSRP standard defines modes and window states. Additional modes can be introduced at developer's risk. A mode states a way of behaving. Depending on the portlet mode, the portlet renders different content and performs different activities.

## 3 Portlet Reusability Model

In this work we are attempting to obtain the dimensions that affect portlet reusability. Reusability is "the degree to which a software module or other work product can be used in more than one computing program or software system" (IEEE, 1990). But portlet reusability is "the capability of the portlet to be used in different portals by several developers". It is important not to get reusability mixed up with interoperability. In the context of WSRP-compliant portlets, judging interoperability does not make sense as these portlets are interoperable by definition. Our model is based on the component reusability model of (Washizaki et al., 2004). In this model,

component reusability is broken down into three dimensions: understandability, adaptability and portability (selected from the ISO/IEC 9126 standard, (ISO, 2001) and adapted on the basis of an analysis of the activities carried out when reusing a black-box component. When reusing a portlet, we carry out the same activities as when reusing a component, hence, the dimensions that affect reusability are the same in both cases. However, these dimensions need to be adapted to account for the peculiarities of the portlet notion. In (Washizaki et al., 2004) adaptability is considered to be a dimension of reusability. However, the ISO/IEC 9126 standard contemplates adaptability as a sub-dimension of portability. Therefore in line with the ISO/IEC 9126 standard, we consider adaptability as a sub-dimension of portability. Furthermore, it should be noticed that reusability can be seen from two points of view: the perspective of the end-user - the person who uses the portlet (end-user issues)- and that of the portal administrator - the one who integrates the portlet into the portal (portal administrator issues)-. Therefore reusability, and hence understandability and portability, must be studied from both perspectives (table 1).

**Table 1.** Dimensions and sub-dimensions proposed for reusability

<i>CHARACTERISTIC</i>	<i>DIMENSION</i>	<i>SUB-DIMENSIONS</i>	
		<i>End-user issues</i>	<i>Portal Administrator issues</i>
Reusability	Understandability	-----	-----
	Portability	Customizability	Installability
		Personalizability	Empathy
			Replaceability

### 3.1 Study of the Reusability Dimensions

Basing our study on the characteristics of a portlet and taking into account the dimensions described for portlet reusability and the two stakeholders identified for a portlet, we are going to present in this section, several attributes for reusability.

**UNDERSTANDABILITY** (table 2) affects portlet reusability because we must understand a portlet’s functionality in order to reuse it: the end-user needs to understand it to be able *to use it*. The portal administrator needs to understand the portlet functionality in order *to integrate it* correctly into the portal.

**Table 2.** Attributes identified in the understandability dimension for each role

<i>DIMENSION</i>	<i>ATTRIBUTES</i>		
	<i>End-user issues</i>	<i>Portal Administrator issues</i>	
Understandability	User manuals	Documentation	
	Description	Documentation language	
	<i>Preview mode</i>		Keywords
			Data types of properties

**Table 3.** Attributes proposed for the sub-dimensions of portability

<i>Dimension</i>	<i>Point of view</i>	<i>Subdimensions</i>	<i>Attribute</i>	
Portability	End-user issues	Customizability	<i>Edit mode</i>	
		Personalizability	Complexity user profile	
			Locales	
	Portal Administrator	Installability	Documentation	
		Empathy	<i>Mime-Type</i>	
			Modes	<i>wsrp:config</i>
				<i>custom modes</i>
			WindowStates	<i>wsrp:minimized</i>
				<i>wsrp:maximized</i>
				<i>wsrp:solo</i>
				<i>custom</i>
				<i>windowStates</i>
			Locales	
			<i>clonePortlet/setPortletProperties</i>	
		CSS		
Replaceability	"mode" ad hoc			
	"extensions" ad hoc			
	"Window State" ad hoc			

Attributes for both end-user and portal administrator are described as follows:

- User manuals: related to the existence of user manuals that help the end-user to understand the portlet easily.
- 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*.
- Preview mode: in this mode, the portlet should provide a rendering of its standard *wsrp:view* mode content, as a visual sample of how this Portlet will appear on the end-user's page with the current configuration.
- Documentation: the portlet provides information related to the portlet that can help the portal administrator to understand the portlet.
- Documentation language: the documentation is provided in several languages.
- Keywords: it refers to the keywords that describe the portlet.
- Data types of properties: it is advisable to obtain information related to the data types of properties, because the more complex data types are, the more difficult it is to understand the properties. This information can be obtained through the method *getPortletProperties*.

**PORTABILITY** affects portlet reusability because a portlet must be capable of being transferred from one environment to another without problems. In Table 3, we set out the subdimensions and attributes that affect portability.

The following subdimensions for portability have been proposed:

- Customizability: capability of the portlet to adapt to itself to the end-user.
- Personalizability: capability of a portlet to allow the end-user to adapt it to his/her preferences.
- Installability: capability of the portlet to be installed.

**Table 4.** Measures proposed for the attributes of the understandability dimension

Dimension: UNDERSTANDABILITY			
Attribute		Measure	Measure domain
End-user issues	User manuals	The portlet vendor provides user manuals.	Boolean (True/False)
	Description	The portlet specifies its functionality.	Boolean (True/False)
	<i>Preview mode</i>	The portlet supports <i>preview mode</i>	Boolean (True/False)
Portal Administrator issues	Documentation	The portlet vendor provides additional documentation.	Boolean (True/False)
	Documentation language	Number of languages in which the documentation is written.	Natural number
	Keywords	Number of keywords provided by the portlet.	Natural number
	Data types of properties	Complexity of the data types of properties	Val.: 3=non-complex, 2=normal, 1= complex

- Empathy: capability of the portlet to harmonise with other independent portlets, both aesthetically and functionally.
  - Replaceability: capability of the portlet to be used instead of another portlet for the same purpose and in the same environment.
- The meaning of the attributes is explained as follows:
- Edit mode: the portlet provides content and logic that let a user customize the behaviour of the Portlet.
  - Complexity user profile: information stored on the portlet related to the end-user.
  - Locales: this is, the different languages that the interface supports.
  - Documentation: that specifies the additional requirements for running the portlet.
  - Mime-Type: mime types supported by the portlet.
  - Modes: modes that the portlet supports for this mime-type. A portlet will be more adaptable if it supports the modes: Wsrp: config (the *config* portlet mode should be used by the portlet to display one or more configuration views that allow administrators to configure portlet preferences. It is defined in the JSR 168 standard) and Custom modes (it allows the declaration of additional custom modes)
  - windowStates: it refers to additional window states that the portlet supports for this mime-Type. They can be: wsrp:minimized (the portlet should not render visible markup, but is free to include non-visible data) , wsrp:maximized (the portlet is likely to be the only portlet being rendered in the aggregated page, or to have more space compared to other portlets in the aggregated page), wsrp:solo (the portlet is the only one being rendered in the aggregated page) and custom windowStates (it allows additional custom window states to be declared).
  - locales: the locale of the returned markup, that is, the language in which returned markup is written.
  - clonePortlet/setPortletProperties: this refers to the chance of creating a new portlet from a existing portlet and of modifying the properties of the cloned portlets.

**Table 5.** Measures proposed for portability dimension

<i>Dimension: PORTABILITY</i>						
<i>Sub-dimensions</i>		<i>Attribute</i>		<i>Measure</i>	<i>Measure domain</i>	
End-user iss.	Customizability	Edit mode		The portlet supports <i>edit</i> mode	Boolean (T/F)	
	Personalizability	Complexity of the user profile		Number of user profile characteristics that the portlet stores.	Natural number.	
		Locales		Number of locales supported by the portlet	Natural number	
Portal Administrator issues	Installability	Documentation		Exist documentation	Boolean (T/F)	
	Empathy	<i>mime-Type</i>		Number of different mime-Type supported.	Natural number	
		Modes	<i>wsrp:edit</i>		The portlet supports <i>edit</i> mode/custom modes	Value among 0 and 2 <sup>2</sup>
			<i>custom modes</i>			
		Window-states (WS)	<i>minimized</i>		Number of additional window states supported by the portlet.	Natural number
			<i>maximized</i>			
			<i>solo</i>			
		locales		Number of locales supported by the portlet		Natural number
	<i>clonePortlet/setPortletProperties</i>		The portlet implements <i>clonePortlet()</i> method and <i>setPortletProperties()</i> method.		Boolean (T/F)	
	CSS		The portlet supports style sheets.		Boolean (T/F)	
Replaceability	"Mode" ad hoc		Number of additional modes supported by the portlet.	Natural number		
	"Extensions" ad hoc		Number of structures extended by the portlet.	Natural number		
	"Window State" ad hoc		Number of additional window states supported by the portlet	Natural number		

- CSS: a common style sheet permits a common look-and-feel across the portlets contained on the page.
- "mode" ad hoc: additional modes exist. Additional modes reduce the portlet replaceability.
- "extensions" ad hoc: extensions exist for the structure.
- "window state" ad hoc: additional window states exist. These additional window states are not mandatory.

### 3.2 Measures and Values

Now that we have identified the attributes that affect the reusability, we can define measures for each one of them. In Table 4, we show the measures defined for the

<sup>1</sup> The measure "the portlet supports *edit* mode/custom modes" can take the value 0 when the portlet does not support any of these modes, 1 when the portlet supports one of these modes and 2 when the portlet supports both modes.

**UNDERSTANDABILITY** and in Table 5 for the **PORTABILITY** dimension together with their domain.

#### 4 Assessment of the Portlet Reusability

Once the measures for each one of the identified attributes have been determined, the portlet reusability level must be ascertained. To do this, we first need to assess the portlet understandability and portability levels.

In order to assess the portlet **UNDERSTANDABILITY LEVEL** (Table 6), we have defined a tuple of seven elements.

**Table 6.** Values proposed for portlet understandability

<i>Understandability</i>	<i>Set of values</i>						
	user manuals	functionality	preview mode	additional documentation	Number of languages supported	Number of key words	Complexity of the data types
Excellent	1	1	1	1	2 <sup>+</sup>	4 <sup>+</sup>	3
High	1	1	1	1	2	[2-3]	[2-3]
Middle	1	1	1	1	1	[2-3]	[2-3]
Acceptable	1	1	0	1	1	2 <sup>+</sup>	1 <sup>+</sup>
Non-accept.	Rest						

Each element of the tuple represents one of the measures presented in Table 4 (in the same order). Having done this, we assess the portlet *understandability* level as this depends on the values of each vector's element. We have taken into account the following considerations in order to assess the values: first, the Boolean measures acquire the value one if they are true and zero if they are false, second, the maximum number of languages in which the documentation is written will normally be two but we also consider the possibility of obtaining a larger number, third, although the number of keywords provided by a portlet is a natural number, we consider that portlets provide 4 keywords, at most; finally, the data types of properties are non-complex and the measure acquires the value 3. When we specify a “+” as superscript of the number, the value for this element is this or greater.

From the **PORTABILITY** point of view, we first need to assess the level of each one of the sub-dimensions which have this as their end: customizability, personalizability, installability, empathy and replaceability.

In the **customizability sub-dimension** we have defined only one measure (Table 5) that will take the value 1 if the portlet supports the edit mode (very acceptable level) and 0 if not (non-acceptable level).

In the **personalizability sub-dimension**, we have established two different measures (Table 5). As a following step, we have assessed the different personalizability levels (Table 7).

**Table 7.** Personalizability levels

<i>Personalizability levels</i>	<i>Set of values</i>
Very acceptable	(22 <sup>+</sup> ,1); (18 <sup>+</sup> ,2 <sup>+</sup> )
Acceptable	([17-21],1); ([14,17],2 <sup>+</sup> )
Non-acceptable	Rest

Therefore *Very acceptable* is when the portlet stores 22 or more user profile characteristics (that is, more than one-quarter of the total) and it supports one locale, or the portlet stores 18 or more user profile characteristics (that is, more than one-fifth of the total) and it supports two or more locales, *Acceptable* is when the portlet stores between 17 and 21 user-profile characteristics (that is, between one-fifth and one-quarter of the total) and it supports one locale, or the portlet stores among 14 and 17 user profile characteristics (that is, between one-sixth and one-fifth of the total) and it supports two or more locales and *Non-acceptable* is the rest of the cases.

In the *installability sub-dimension*, we have defined only one measure (Table 5) that acquires the value 1 (very acceptable level) when the portlet provides documentation related to the installation and 0 in other case (non-acceptable level).

In the *empathy sub-dimension*, we have set out the measures presented previously (Table 5) and have taken into account the following considerations to assess the level of empathy: first, normally the maximum number of mime-type supported by the portlet is 2 (nevertheless, we consider the possibility of obtaining a greater number), second, we consider that the measure “number of additional window states supported by the portlet” normally acquires a value between 0 and 4 (in the worst case, the portlet does not implement any additional window state and in the best case, the portlet implements the 4 additional window states. It is also possible that the portlet may provide more than one *custom* mode. We consider this possibility, but it is not usual), third, the usual maximum number of languages supported by the interface is 2 (in spite of this, here too we consider the possibility of obtaining a larger number), finally, the measure “the portlet implements *clonePortlet()* method and *setPortletProperties()* method” acquires the value one when it implements both methods and it acquires the value zero when it does not implement any method.

Table 8 outlines the output where each level is characterised as follows: *Very acceptable* (when the portlet supports two mime-types, it supports the *edit* mode and it defines one or more *custom* modes. It also supports the *minimized*, *maximized*, *solo* window states. The portlet defines one or more *custom* window states and it can be rendered in one or more locales. It implements the *clonePortlet()* and *setPortletProperties()* methods and it supports style sheets.), *Acceptable* (if the portlet supports one mime-type; it supports the *edit* mode but it does not define *custom* modes. The portlet supports the *minimized*, *maximized*, *solo* window states, but it does not define *custom* window state. The portlet can be rendered in one or more locales; it implements the *clonePortlet()* and *setPortletProperties()* methods and it supports style sheets) and *Non-acceptable* (all other cases).



**Table 8.** Empathy level

<i>Empathy level</i>	<i>Set of values</i>					
	number of different mime-Type supported	the portlet supports <i>edit</i> mode/ <i>custom</i> modes	number of additional window states supported by the portlet	number of locales supported by the portlet	the portlet implements <i>clonePortlet()</i> method and <i>setPortletProperties()</i> method	the portlet supports style sheets
Very Acc.	2	2 <sup>+</sup>	4 <sup>+</sup>	1 <sup>+</sup>	1	1
Acceptable	1	1	3	1 <sup>+</sup>	1	1
Non-Acc.	Rest					

**Table 9.** Replaceability levels

<i>Replaceability level</i>	<i>Set of values</i>		
	number of additional modes supported by the portlet	number of structures extended by the portlet	number of additional WS supported by the portlet
VeryAcceptable	0	0	0
Acceptable	Rest		
Non-acceptable	4 <sup>+</sup>	1 <sup>+</sup>	4 <sup>+</sup>

We have taken into account the considerations seen below in assessing the *replaceability sub-dimension* level (Table 9): first, for the measure “number of additional modes supported by the portlet” (Table 5): if the portlet supports additional modes it reduces its replaceability, because the consumer may not support additional modes, second, for the measure “number of extended structures by the portlet”: if the portlet extends the predefined structures, it reduces its replaceability because it is possible that the consumer does not support the extended structures, finally, for the measure “number of additional window states supported by the portlet”: if the portlet supports window states different to *normal* window state it reduces its replaceability, because the consumer may not support additional window states.

Where: *Very acceptable* (means that the portlet only supports the *view* mode; it does not extend the predefined structures and it only supports the *normal* window state), *Acceptable* (means the rest of cases) and *Non-acceptable* (means that the portlet either supports all the additional modes (*edit*, *help* or *preview*) but does not define *custom* modes or, on the other hand, supports some additional modes and defines *custom* modes. The portlet extends one or more structures. The portlet supports all the additional window states: *minimized*, *maximized* or *solo* but it does not define *custom* window states or, on the other hand, supports some additional window states and defines *custom* window states.).

Once we have determined the portlet level for each of the sub-dimensions, we could assess the portlet **PORTABILITY LEVEL** on the basis of the obtained values for each sub-dimension. The different portability levels are summarized in Table 10<sup>2</sup>.

<sup>2</sup> Two of the five sub-dimensions which make up the portability, acquire the *very acceptable* or *non-acceptable* value. For this reason, the portability level is *acceptable* when those sub-dimensions obtain the *very acceptable* value.

We can assess the portlet **REUSABILITY LEVEL** according to the understandability and portability level that the portlet has obtained. The different levels defined for the portlet reusability are set out in Table 11.

**Table 10.** Portability levels

<i>Portability levels</i>	<i>Set of values</i>
Excellent	All the sub-dimensions have obtained the <i>very acceptable</i> level.
High	One sub-dimension has obtained acceptable and the rest the <i>very acceptable</i> level.
Middle	Two sub-dimensions have obtained <i>acceptable</i> and the rest the <i>very acceptable</i> level.
Acceptable	Three sub-dimensions have obtained <i>acceptable</i> and the rest the <i>very acceptable</i> level.
Non-acceptable	Rest of the cases.

**Table 11.** Reusability levels

<i>Reusability level</i>	<i>Set of values</i>
Excellent	The two dimensions (understandability and portability) have obtained the <i>Excellent</i> value.
High	One dimension has obtained the <i>Excellent</i> value and the other has obtained the <i>High</i> value. One dimension has obtained the <i>Excellent</i> value and the other has obtained the <i>Middle</i> value. The two dimensions have obtained the <i>High</i> level.
Middle	One dimension has obtained the <i>Excellent</i> value and the other has obtained the <i>Acceptable</i> value. One dimension has obtained the <i>High</i> value and the other has obtained the <i>Middle</i> value. One dimension has obtained the <i>High</i> value and the other has obtained the <i>Acceptable</i> value. The two dimensions have obtained the <i>Middle</i> value.
Acceptable	One dimension has obtained the <i>Middle</i> value and the other has obtained the <i>Acceptable</i> value. The two dimensions have obtained the <i>Acceptable</i> value.
Non-Acc.	Rest of cases.

## 5 Applying the Reusability Model for Portlets to a Specific Portlet

In order to illustrate our portlet reusability model, we have applied it to a specific portlet. The goal of this portlet is to carry out a search for cars, but it is not a commercial portlet. It is used for research and more specifically, for performing tests. For this reason, part of its capabilities is not implemented and the portlet is not available to the rest of the community. This portlet has been developed using the JSR168 standard. Table 12 shows the results for the **UNDERSTANDABILITY** dimension. Following our model and taking into account the values shown in Table 12, the portlet understandability is not acceptable.

**Table 12.** Results for understandability

<i>Measure</i>	<i>Value</i>
The portlet vendor provides user manuals.	False
The portlet specifies its functionality.	True
The portlet supports <i>preview</i> mode	False
The portlet vendor provides additional documentation.	False
Number of languages in which the documentation is written.	0
Number of keywords provided by the portlet.	1
Complexity of the data types of properties	1

**Table 13.** Results for subdimensions of portability

<i>Subdimension</i>	<i>Measure</i>	<i>Value</i>
Customizability	The portlet supports <i>edit</i> mode	True
Personalizability	Number of user profile characteristics that the portlet stores.	0
	Number of locales supported by the portlet	1
Installability	Exist documentation	False
Empathy	Number of different mime-Type supported.	1
	The portlet supports <i>edit</i> mode/custom modes	1
	Number of additional window states supported by the portlet.	0
	Number of locales supported by the portlet	1
	The portlet implements <i>clonePortlet()</i> method and <i>setPortletProperties()</i> method.	False
	The portlet supports style sheets.	False
Replaceability	Number of additional modes supported by the portlet.	0
	Number of structures extended by the portlet.	0
	Number of additional window states supported by the portlet	0

We assess the values for the **PORTABILITY** dimension as the next step (Table 13). To do this, we assess the values for each sub-dimension. According to the results, the portlet customizability is very acceptable, the personalizability is not acceptable, the installability is non-acceptable, the empathy is non-acceptable and the replaceability is very acceptable.

Therefore, taking into account levels of the portability sub-dimensions, the portlet portability is non-acceptable because, although there are two sub-dimensions of portability that have obtained the value of very acceptable, there are three that have obtained the value non-acceptable.

Finally, according to the portlet understandability and portability levels, we can affirm that the portlet **REUSABILITY** is non-acceptable. This result is coherent because the portlet is not a commercial portlet and as such it does not implement all the possible characteristics. It does not provide additional documents which are necessary to obtain a good value for portlet reusability. Having assessed the portlet reusability level we are in a position to improve the obtained reusability level, implementing those characteristics necessary for portability and understandability dimensions to obtain a higher level such the inclusion of user manuals, or the implementation of the *clonePortlet* and *setPortletProperties* methods.

## 6 Conclusions and Future Work

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. The proposal and ample support for the WSRP portlet standard predict an emergent portlet market. A main requirement for the blooming of this market is the existence of portlet quality models that assess portal developers in selecting the appropriate portlet.

This work provides some insights on a reusability model for portlets based on the WSRP standard. As a first attempt, the model has been applied to a sample portlet. The output serves not only to indicate the adequacy of the portlet but to guide the portlet developer to detect the weaknesses of the portlet as far as reusability is concerned.

In future work, we have to validate through surveys done by experts, both the dimensions and the measures proposed for the usability, but especially the set of values used to assess the portlet usability.

Also, we plan to define other models for the rest of the portlet quality characteristics until we will have dealt with all characteristics that affect that quality. These characteristics are: functionality, reliability and efficiency.

The final goal is to have a quality model that could be used, on one hand, to decide on the best portlet and, on the other hand, to identify possible improvements in the quality of a given portlet.

## Bibliography

- Collins, D. (1999). Datawarehouses, enterprise information portal and the SmartMart meta directory. *Information Builders Systems Journal*. Vol. 12(2) pp. 53-61.
- Diaz, O., C. Calero, M. Piattini and A. Irastorza (2004). Portlet usability model. IBM Research Report. RA221(W0411-084). ICSOC 2004.,pp. 11-15.
- IEEE (1990). Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries., New York
- ISO (2001). ISO/IEC 9126. Software Engineering-Product Quality. Partes 1 a 4., International Organization for Standardization/International Electrotechnical Commission.
- Kvitka, C. (2002). Profiting with portals. May/June. 21. Available in: [http://www.oracle.com/technology/oramag/oracle/02-may/o32news\\_portals.html](http://www.oracle.com/technology/oramag/oracle/02-may/o32news_portals.html). Accessed: Septiembre, 2004.
- OASIS (2005). Web Service for Remote Portals (WSRP) Version 2.0. Available in: <http://www.oasis-open.org/committees/wsrp/>. Accessed: June.
- Washizaki, H., H. Yamamoto and Y. Fukazawa (2004). A Metrics Suite for Measuring Reusability of Software Components. Software Metrics Symposium. Ninth International.,pp. 211-223.