# Pedagogical Engineering Fundamentals to Build Robust Software Components for Online Laboratories

Bouchaib FATTOUH; Hamadou SALIAH-HASSANE
Télé-Université & LICEF/CIRTA Research Centre Montréal
bfattouh@licef.teluq.uquebec.ca, saliah@teluq.uquebec.ca

KEYWORDS: *graphical user interface designer, pedagogical engineering, cognitive engineering, information systems engineering, telelaboratory.*

ABSTRACT: *This paper presents a strategy used in our research to build a distributed user interface designer. In this case, the interfaces are specific learning objects used in synchronous online laboratory activities. In this perspective, our approach is based on the fundamentals of pedagogical engineering methodology, which is the association of processes and principles of the three domains known as pedagogical design, information systems engineering and cognitive engineering. In order to show the usefulness of our approach we used a methodology called MISA, a learning system engineering methodology developed at the LICEF/CIRTA Research Centre in the framework of a research project called Learning Object Repository Network (LORNET).*

## 1 INTRODUCTION

In earlier papers, while addressing the topic of telelaboratory development as a networked learning system, we introduced the distributed interface generator designer as one of the important environments and a key element of the telelaboratory. This environment is intended to help designers query a server for a graphical user interface and configure it to control measuring instruments. In this case, the user's interfaces are specific learning objects that correspond to laboratory devices used in synchronous telelaboratory activities. As such, the main function of the distributed interface generator development aims to provide designers with a large inventory of capabilities to facilitate searching graphical user interfaces required by an experiment activity, objects mapping to a pedagogical scenario, simulations tasks and experimental procedures settings in collaborative projects while sharing objects with other designers regardless of their remote geographical locations.

However, this motivation involves development efforts based on a specific methodology since the distributed interfaces generator has its own unique set of requirements. Moreover, in addition to technical concerns, pedagogical issues must also be considered. Thus, as it would be unwise to develop this system without first tackling these critical issues, we chose to use the Learning Systems Engineering Method (MISA) developed at the LICEF/CIRTA Research Centre in the framework of a research project called Learning Object Repository Network (LORNET). The models and documented elements produced by this method allowed us to propose a solution that fulfils the development needs of robust telelaboratories software components.

## 2 PEDAGOGICAL ENGINEERING

Successfully used to develop several Web training environments and several courses managed from remote sites [1], MISA, the learning systems engineering method, aims to apply pedagogical and cognitive engineering to construct learning systems [2]. A graphical and structured model is used to represent the method processes, products and principles. Its goals aim to produce a learning system described through four distinct models, namely (a) a learning object model; (b) a pedagogical model comprised of processes or pedagogical scenarios; (c) a media model highlighting the pedagogical materials and (d) a delivery model that addresses the technological infrastructure supporting the delivery of the learning system. Although each segment is modeled separately, and in spite of the fact that at a glance they seem to represent one of three different dimensions, they remain closely related. This representation facilitates linking the four models in order to control the quality of the learning system. This graphical representation is achieved through the use of MOT (*Modélisation par Objets Typés*), that is an object-oriented modeling tool developed at LICEF [3]. This tool utilizes graphic symbols to represent concepts with rectangles, principles with hexagons, procedures with ovals and facts with two overlapping rectangles (See Figure 1).

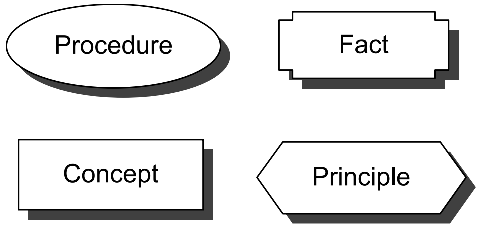


Figure 1 – Graphical representations of different types of knowledge

Links in the format of arrows are used to connect knowledge objects with the pedagogical contents. For instance, the letter **C** represents a composition link, meaning that a knowledge object is made up of another knowledge object. The **I/P-**link means either input or product, depending on the direction of the arrow. A procedure often includes an input concept or a product. Additionally, the procedure can be controlled by a principle represented by an **R-**link that establishes the conditions; the **P-**link specifies that a procedure takes precedence over another. Finally, the **S-**link is used to indicate that a knowledge object is "a kind of" and has the same characteristics as the knowledge object to which it is associated [3].

In addition to the aforementioned four models, MISA adheres to the learning system engineering process that includes six phases: learning problem identification; preliminary design phase; architecture phase; pedagogical material design phase; pedagogical material development phase and the learning system setup. The documentation produced during these processes is used by both the project management team and the programmers who develop the software components of the learning system [3].

## 3 THE GRAPHICAL USER INTERFACE GENERATOR

Now that the main concepts of MISA have been introduced and that its flexibility has been highlighted, we select the models susceptible to fulfill our main goal: to develop the distributed interface generator that will be used by pedagogical designers to produce learning scenarios using a wide variety of materials and learning objects. Moreover, it is important to specify that these materials are not produced by MISA since this method aims to manage detailed descriptions of the leaning system, the requirements and the nature of the learning objects as well as its set of properties and attributes.

Moreover, the MISA models described above are useful to resolve pedagogical concerns and to gather the requirements and features that the learning system should contain. These requirements are then passed to actors such as project managers and eventually programmers who produce the learning system materials. From these processes and requirements stem the need of an interface generator system, that is a tool, compatible with other resources and materials, possibly used in conjunction with a function editor that conducts pedagogical engineering tasks.

### 3.1 The knowledge model

The interface generator designer is an environment that facilitates generating the graphical user interfaces needed to complete telelaboratory activities. We have already stated that the knowledge model is a way of illustrating the objectives of such activities in terms of the knowledge, skills and abilities that a learner will gain after completing a given laboratory activity. Figure 2 illustrates a model representation produced with the MOT software. In this example, an activity designed for future electrotechnicians aims to give learners hands on experience in measuring direct current while using specific instruments, such as a power supply, a digital multimeter and an oscilloscope.

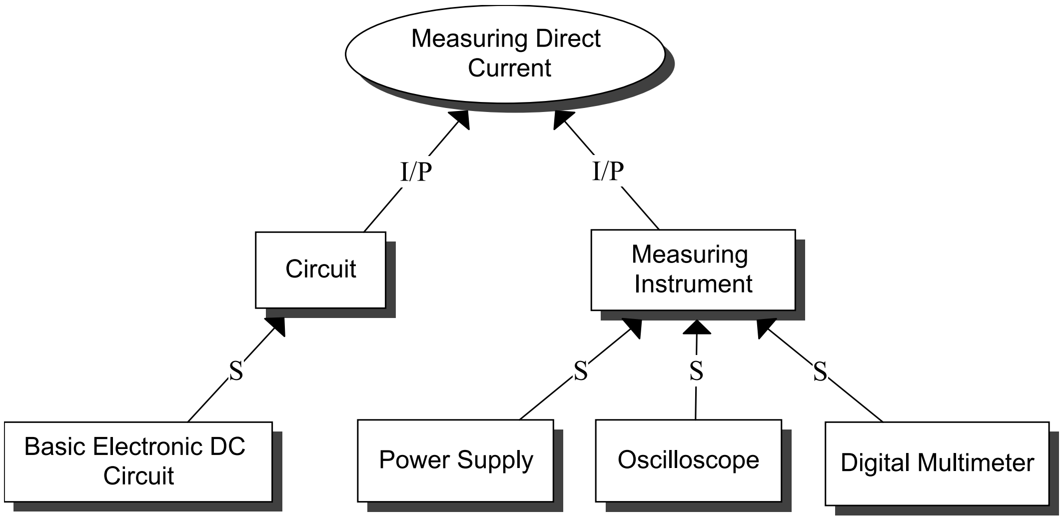


Figure 2 – The knowledge model

### 3.2 The pedagogical model

This model, represented with the same graphical symbols using MOT, describes the learning process based on the knowledge model. It illustrates how the laboratory activity and procedures are conducted, as well as the interactions amongst the actors and objects. It exhibits the fact that objects are distributed, shared by multiple learners regardless of their remote geographical locations. Figure 3 illustrates the pedagogical model.

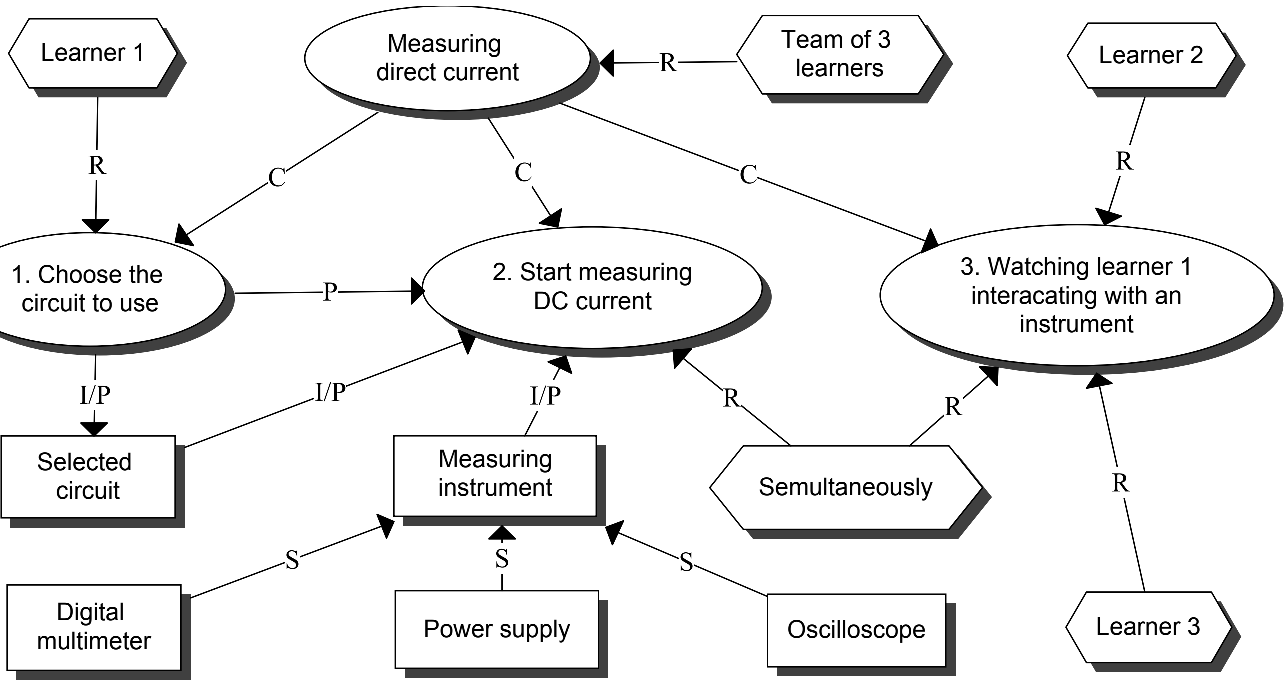


Figure 3 – The pedagogical model

### 3.3 The media model

The media model describes the nature of the teaching materials previously selected with the knowledge and the pedagogical models. Based on these models we can easily identify the learning objects and eventually the graphical user interfaces needed to complete a laboratory activity. MISA is helpful in that it suggests the production of a document with the critical attributes of all objects, taking into account the specific characteristic which is the synchronous use of the telelaboratory learning objects in a collaborative environment.

Since the telelaboratory learning objects are specific in that they should map to the corresponding networked devices to work properly, an important issue raises to that extent. In fact, learning objects must be compatible with standards such as the meta-data provided by the IEEE LOM [4], yet the latter is not sufficient and indeed there is a need to refine the relationships expressed among the objects and eventually other required resources likewise the ones needed in the course of telelaboratories. IMS LD suggests a meta-model to describe relationships between objects through building a framework that defines the structure for the content and the behaviour of the different types of learning objects [5].

## 4 THE GRAPHICAL USER INTERFACES DESIGN

A digital multimeter is one of the learning objects retrieved by our models. It is a typical measuring instrument which is widely used in practical courses, a useful source of information found for this specific context is the IVI (Interchangeable Virtual Instrument) foundation [6], an open consortium founded to promote specifications for instruments measurements. The working groups of this consortium made available an IVI class for digital multimeters (DMMs) covering specifications for typical DMMs as well as for complex instruments. Using these specifications was an important step before launching the interface production (See Figure 4).

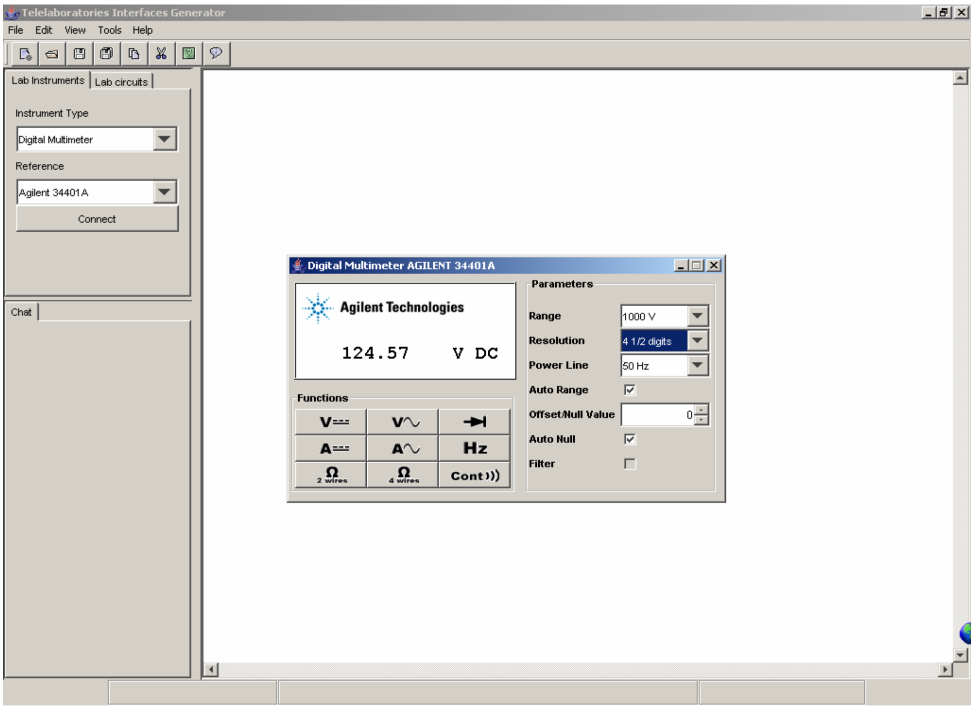


Figure 4 – The graphical user interface generator

The graphical user interfaces generator shows the Agilent 34401A [7] interface, we developed as well the NI 4060 [8] shown below, in Figure 5. In addition to digital multimeters, IVI defined other class specifications corresponding to instruments such as oscilloscopes, function generators, DC power supplies, switches, power meters, spectrum analyzers and RF signal generators.

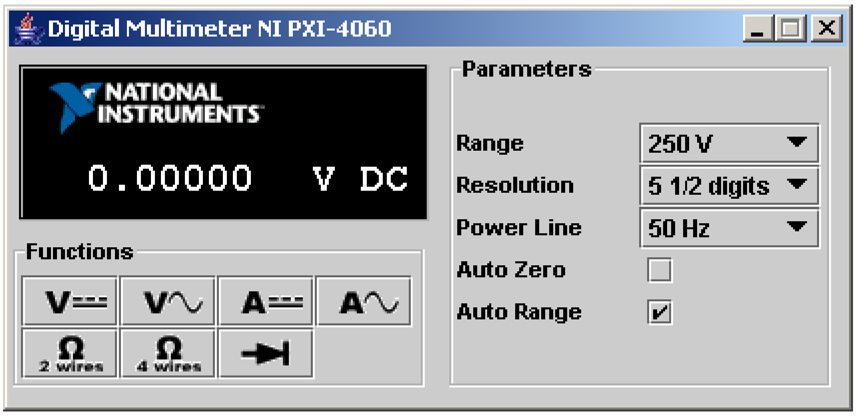


Figure 5 – The NI 4060 DMM graphical user interface

## 5 THE GRAPHICAL USER INTERFACE ARCHITECTURE

The technological solution we propose is sufficiently modular, the developed framework is easy to extend and make the graphical user interfaces (GUI) of the entire measurement instruments using the IVI class specifications much like the one shown above. The efficiency of our solution lives in its generative way in generating the interfaces. The same method will be used to develop circuit interfaces such as electronic or electric circuits.

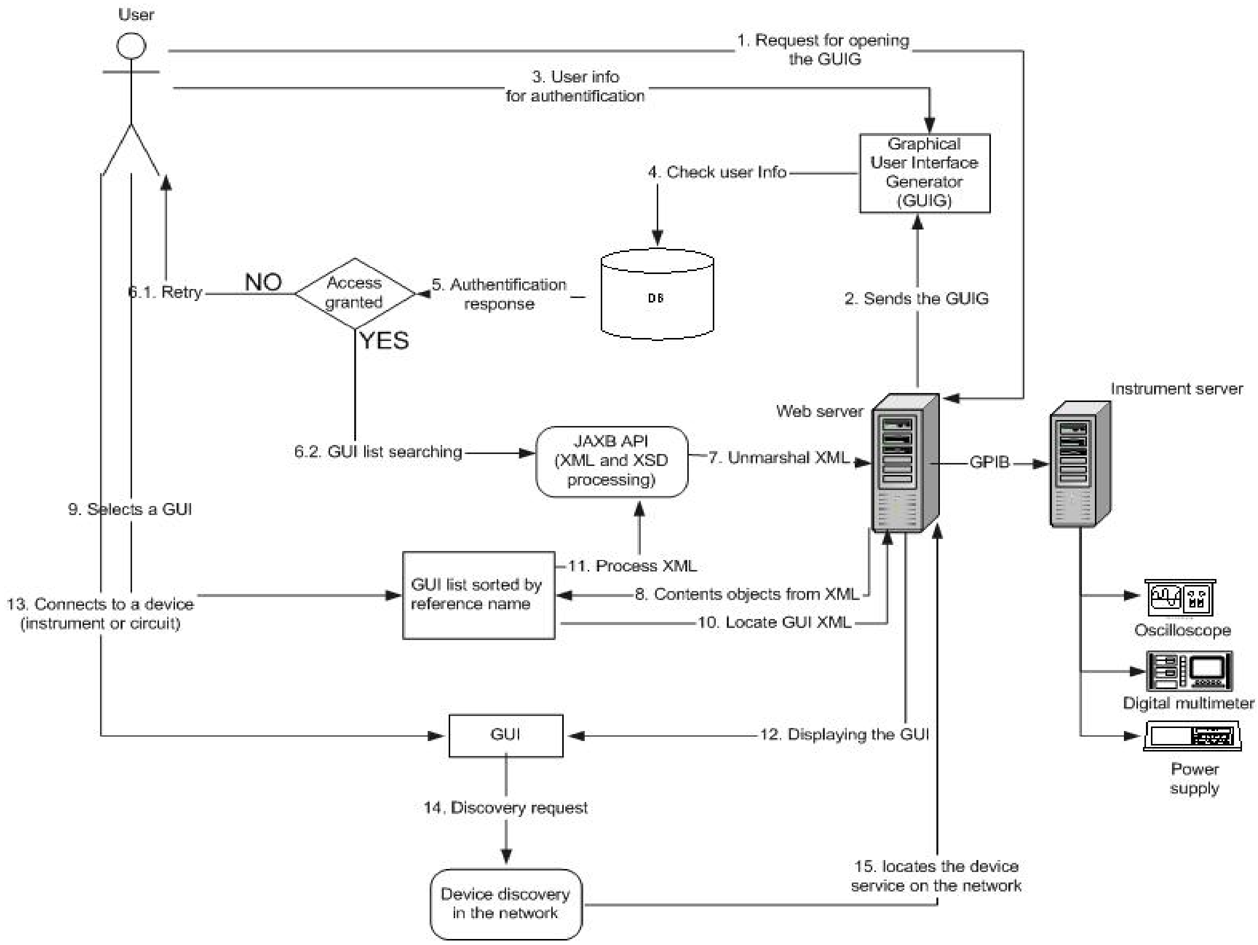


Figure 6 – The graphical user interface generator architecture

Figure 6, shows the use of W3C XML schema language [9] that describes the structural data types corresponding to measurement instruments attributes. This schema is then used to validate an XML (Extensible Markup Language) file containing attributes values such as function, range, resolution and so forth. To access the XML document, we used JAXB (Java Architecture for XML Binding) [10], this approach allows coupling both partners XML and the schema language to produce an easy to use API for the XML processing and generating the corresponding GUI which is then distributed among multiple users regardless of their remote geographical locations.

**ACKNOWLEDGEMENTS**

We would like to thank two collaborators, M. Julien Contamines and M. Johnny Villeneuve, who worked closely with us and contributed to our research. This work is funded by the Natural Science and Engineering Research Council of Canada under the Learning Object Repository Network (LORNET) project.

**REFERENCES**

[1] H. H. Saliah, M. Saad & H. Hassan, I. De la Teja. *Virtually and remotely accessing and controlling laboratory real devices: A new trend in teaching and learning in engineering.* Istanbul Turkey: Proceedings of the International Conference on Information Technology Based Higher Education and Training, 2000. 8 s.

[2] Gilbert Paquette. *L'ingénierie pédagogique - Pour construire l'apprentissage en réseau.* Montréal: Presses de l'Université du Québec, 2004. 456 s. ISBN 2-7605-1162-6.

[3] G. Paquette, M. Léonard, I. de la Teja, M. Paule Dessaint. *Méthode d'ingénierie d'un système d'apprentissage MISA 4.0 Présentation de la méthode*. Montréal : LICEF, 2001. 68 s.

[4] Wayne Hodgins. Standards for Information Technology-Education and Training Systems-Learning Object and Metadata. IEEE Learning Technology Standards Committee, December 2002. Available from web: <URL: http://ltsc.ieee.org/wg12/index.html>

[5] IMS Global Learning Consortium, Inc. IMS Learning Design Best Practice and Implementation Guide, October 2002. Available from web: <URL: http://www.imsglobal.org/learningdesign/>

[6] IVI Foundation, IVI-4.2: IviDmm Class Specification, March 2002. Available from web: <URL: http://www.ivifoundation.org/ApprovedSpecs/IVI-4.2_DMM_v3.0.pdf>

[7] Agilent Technologies, 34401A Multimeter User's Guide, December 1998. Available from web: <URL: http://cp.literature.agilent.com/litweb/pdf/34401-90004.pdf>

[8] National Instrument, NI PXI 4060 Specifications. Available from web: <URL:http://www.ni.com/pdf/products/us/4mi428-430.pdf>

[9] Henry S. Thompson, David Beech, Murray Maloney, Noah Mendelsohn, XML Schema Part 1: Structures, May 2001. Available from web: <URL: http://www.w3.org/TR/xmlschema-1/>

[10] Ed Ort, Bhakti Mehta, Java Architecture for XML Binding (JAXB), March 2003. Available from web: <URL: http://java.sun.com/developer/technicalArticles/WebServices/jaxb/index.html>