A Web-based Interactive Visual Component for Online Collaborative Business Process Management

Aizhong Lin, Mao Lin Huang and Igor T. Hawryszkiewycz
Faculty of Information Technology,
University of Technology, Sydney
P.O.Box 123, Broadway
NSW 2007, AUSTRALIA
{alin, maolin, igorh}@it.uts.edu.au

Abstract

This paper describes a web-based interactive visual component for supporting online collaborative business process management. This component supports not only the outputs of visual information but also the input of visual information interactively. Most of traditional web-based visualizations are only used for viewing information and they are not interactive. They focused on how rich graphics could simplify the cognition process in understanding information, but they paid less attention on how to use visualizations as an interface for input and output of information. Most traditional web-based visual components are unable to send input information from web clients to web servers, especially when network firewalls or proxies exist between the web clients and the web server. The web-based interactive visual component introduced in the paper employs a HTTP-based communication module to transfer information from web clients to the web server. The HTTP-based communication module plays the role of the information management infrastructure that receives the information and manages them in an information repository.

Keywords: collaborative business process management, Web-based visualization, Interactive visualization

1. Introduction

A *business process* is a set of related activities to achieve the business goal. From the management perspective, a business process is represented by a set of process elements (including *goal, activity, role, participant, knowledge,* and so on) and a set of relationships among elements. These process elements and relationships are stored in an information system. To support *online collaborative business processes* (such as a large research project), in which a number of people (called process participants) are involved, process elements and relationships are saved in a web-based information system (such as a database management system situated in a computer installed with a web server) so that process participants in different places can perform the same process collaboratively by using their web browsers [HIT, 1999]. The simple and traditional representation of a business process on the web browser is a list of the element in which the related elements in a business process are listed in a table. The element list is useful showing all process elements in a sequential manner to the participant. However, it does not represent the relationships among elements (such as the "responsible" relationship between a *role* and a process *activity*).

Visualization technology has been widely used in scientific computing [DYMOS, 2003], information retrieving [Card, 1996][HEL, 2003], and data mining [3DAC, 2003]. Researchers applied visual components in information management because the visual components can show not only the information but also the relationships between the pieces of information to the user who retrieves the information. This enables the user to quickly view the information and retrieve the information. The *web-based visualization* applies Hypertext Transport Protocol (HTTP) to transfer information, uses services activated by the HTTP server to manage the information, and uses the web browser to browse the visualized information. Consequently, web-based visualization is used in this research to support the web-based management of collaborative business processes. Most of traditional web-based visualizations are only used for viewing information and they are not interactive. They focused on how rich graphics could simplify the cognition process in understanding information, but they paid less attention on how to use visualizations as an interface for input and output of information. Most traditional web-based visual components are unable to send input date from web clients to web servers, especially when network firewalls or proxies exist between the web clients and the web server.

This paper describes a web-based interactive visual component (WbIVC) for supporting online collaborative business process management. The

WbIVC is built with a HTTP-based communication module that transfers information from a web-server to web clients as well as from web clients to the web server. The use of HTTP communication module ensures that the data is not blocked by firewalls or proxies. The WbIVC is employed in a collaborative business process management system to provide interactive interfaces for process participants.

2. The Collaborative Business Process Meta-Model

A business process meta-model defines the process elements and the relationships between the process elements. Figure 1 shows the meta-model of business processes. A business process consists of *goal, activity, role, participant, artifact, patron,* and *permission.* The relationships between the elements are *is_a, has, fills, sets, accesses, performs, achieves,* and *constraints.* The process elements and their relationships are defined as follows. As shown in figure 1, the meta-model centers on a business process. A patron, it is a specific rôle, is responsible for the business process from its creation to its termination. Others roles have their responsibilities and the permissions set by the process patron. People, here called participants, fill the roles so they take the responsibilities to achieve goals by performing activities. The constraints are used when performing activities. The meta-model illustrated here is collaborative because many roles and participants may be involved in a business process.

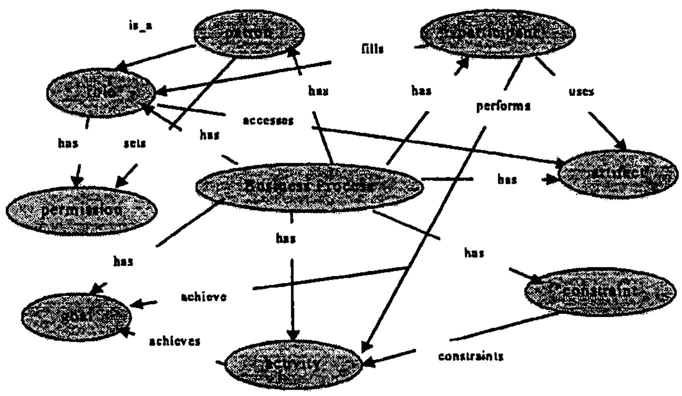

Figure 1: The meta-model of a business process

- *goal* A goal is a description that the process patron intends to achieve
- *patron* A patron is a specific role who is responsible for the business process from its creation to its termination
- *role* A role is an abstract position consists of responsibilities
- *participant* A participant is a specific person assigned to a role or roles
- *activity* An activity is a description of a piece of work that forms one logical step within a process. It produces a well defined output and usually require many actions and interactions to do so
- *artifact* An artifact is a data object such as documents
- *action* A specific unit of work carried out by a role (eg. Change an artifact, send an artifact)
- *permission* A permission is a set of signals that defines a participant is allowed or not allowed to access a artifact
- *constraint* A constraint is a condition which must be met during a process
- *is_a* The "from" object is a part of the "to" object. For example, the patron is a specific role
- *has* The "from" object has the "to" object as a part. For example, a role has a set of permissions to access artifacts.
- *fills* The "from" object takes the position of the "to" object
- *sets* The "from" object gives values to the "to" object
- *accesses* The "from" object operates on the "to" object according to the permission
- *performs* The "from" object does the works describes in the "to" object
- *achieves* The "from" object tries to realize the "to" object
- *constraints* The "from" object has some level of control to the "to" object

Figure 2 is a traditional human-computer interface to display the business processes in a business process management for a specific process participant. The research process list shows all research projects for the participant "alin". When a research project is opened, a new list shows the project elements and their relationships. The display is not visualized and the participants have to spend plenty of time to understand which element is related to another element according to a relationship.

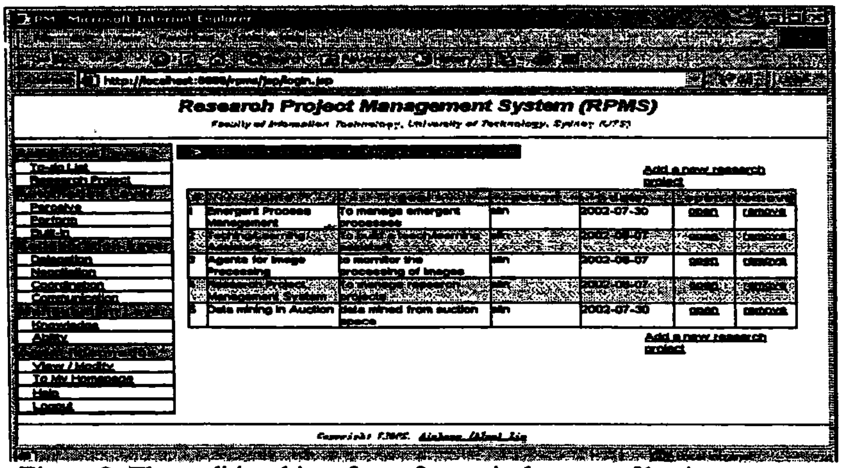

Figure 2: The traditional interface of a particular type of business process management system --- Research Project Management System

3. The WbIVC

To provide more efficient tools for process participants to view and review the business processes (especially the collaborative business processes) so that the process can get quickly responded, a web-based visualization component is developed for business process management system. Figure 3 illustrates the component architecture. In the server side, three modules --- a web-service, a visualization module, and an XML [XML, 2000]-based information repository --- are built in the WbIVC. The HTTP-based communication module service is built to transfer information between the web server and web clients and then it manages the information which are represented and stored in a XML-based information repository. In this research, the information repository is used to represent and store business process elements and relationships. The visualization module is implemented in a Java applet which provides the rich graphical output and input interfaces for process participants. A process participant can use a web browser to access the business processes via the Internet. The applet-based rich graphical interfaces are shown in the web browser.

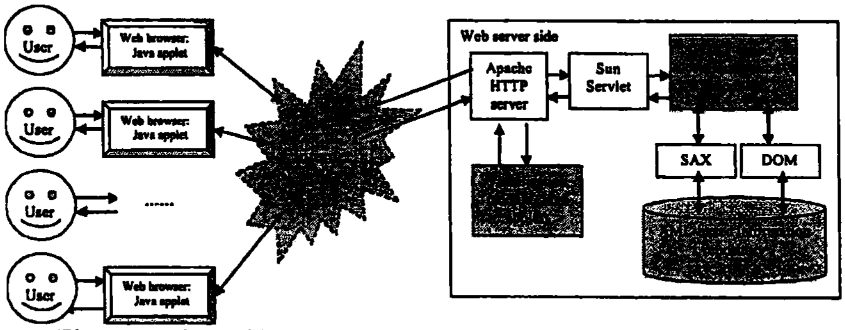

Figure 3: The architecture of the web-based visualization component

3.1 The XML-based Information Repository

In the web server side, business processes are represented by using process elements and relationships and stored in an XML-based information repository. The information repository has a three levels tree structure as shown in Figure 4. The top level of the tree is the name of the business process repository (Eg. bpr). It is realized by creating a folder named "bpr" under the web server folder. The middle level represents the name of each business process. The leaf level stores the business process elements and relationships as .xml files.

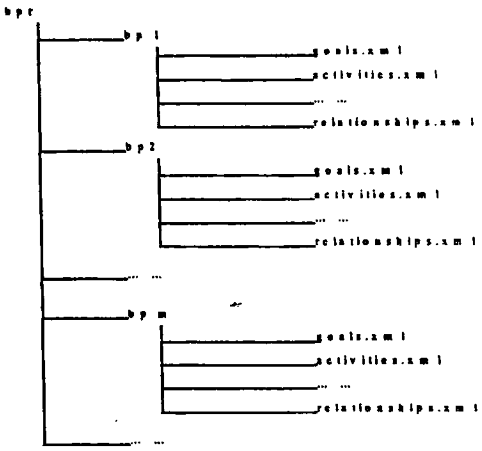

Figure 4: The three-level tree structure of a business process repository

When a process participant (say "alin") retrieves a specific business process, alin sends a request to the web server. In the server side, according to this request, all the .xml files in the folder (the folder name is the business process name) is read and then sent to the alin. In contrast, if alin sends a business process to the server, all its elements and relationships are saved in the folder as .xml files.

3.2 The HTTP-based Communication Module

The HTTP-based communication module realizes two types of transformation of information between the web server and web clients. The first communication function is similar as the normal HTTP by which the client sends a request to the server and the server retrieves the information according the request and then sends the information to the client. Figure 6 and Figure 7 is the major source of the request and response respectively. The code in Figure 6 is situated in the Java applet which is executed in the clients' web browser. The code listed in Figure 7 is executed in web server.

```
public static String sendRequestToServer(String request)
{
    try
    {
        BufferedReader  br   = null;
        StringBuffer    sb = new StringBuffer();
        InputStream     in   = null;
        URLConnection   conn = null;
        URL             url  = null;
        String          line = null;

        url = new URL("http://138.25.13.183:8888/pwm/servlet/RPMSServlet"+ request);
        conn = url.openConnection();
        conn.setUseCaches(false);
        in   = conn.getInputStream();

        br = new BufferedReader(new InputStreamReader(in));
        while ((line = br.readLine()) != null)
        {
            sb = sb.append(line + "\r\n");
        }

        ……
    }

    ……
}
```

Figure 6: The source code to send the request from a web client to the web server

```
protected void rRequest(HttpServletRequest request, HttpServletResponse response, String agent)
        throws IOException, ServletException
{
        HttpSession session = request.getSession();

        PrintWriter out = response.getWriter();
        response.setContentType("text/plain");

        ……

        String str = RPMSTools.XMLFileConvertToString(rpName + ".xml", "RPElementDB");

        out.print(str);
        out.flush();
}
```

Figure 7: The source code to reply the request from the web server to the web client

The second communication function supports the client sends the information to the server and the server saves the information to the XML-based information repository according the command situated in the front of the information. And then the server sends a feedback to the client to tell this action is successful or unsuccessful. Figure 8 and Figure 9 is the major source code of the information sending and server's action. Figure 8 is situated in the java applet and Figure 9 is executed in the web server.

```
public static String sendDataToServer(String data)
{
    try
    {
        StringBuffer sb = new StringBuffer();

        URL url = new URL("http://138.25.13.183:8888/pwm/servlet/RPMSServlet");

        URLConnection uc = url.openConnection();
        uc.setDoOutput(true);
        uc.setDoInput(true);
        uc.setAllowUserInteraction(false);

        DataOutputStream dos = new DataOutputStream(uc.getOutputStream());
        dos.writeBytes(data);
        dos.flush();
        dos.close();

        ......
    }

    ......
}
```

Figure 8: The source code to send the information from a web client to the web server

```
protected void getDataFromPost(HttpServletRequest request, HttpServletResponse response)
        throws IOException, ServletException
{
        HttpSession session = request.getSession();
        PrintWriter out = response.getWriter();
        response.setContentType("text/plain");

        ServletInputStream sis = request.getInputStream();

        BufferedReader br = new BufferedReader(new InputStreamReader(sis));
        StringBuffer sb = new StringBuffer();
        String line = "";
        while ((line = br.readLine()) != null)
        {
                sb = sb.append(line + "\r\n");
        }

        ......

        RPMSTools.objectConvertToXMLFile(rp, rpName + ".xml", "RPElementDB");
}
```

Figure 9: The source code to do action according to the command in the front of the information

3.3 The Visualization Module

Formally, a business process *BP* is represented by finite set of process elements *PE* and a finite set of element relationships *ER*, i.e., *BP = {PE, ER}*. In the visualization module, a process element (pe) included in the PE is represented by a colored and labeled rectangle. The label in the rectangle is the name of the process element name. An element relationship (er), which links one pe (pe_a) to another pe (pe_b), is represented by a line with a single arrow (Figure 10). The single arrow line means the pe_a manipulates pe_b. Normally, a colored and labeled rectangle is called a node, and a single arrow line is called an edge. Therefore, a business process graph consists of the nodes and edges (Figure 11).

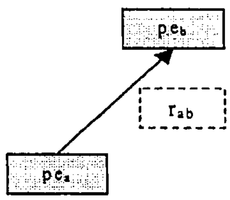

Figure 10: An element relationship links two process elements

When moving the mouse pointer in a node and then clicking the right button, a popup menu is activated. A number of menu items are listed in the popup menu. By clicking one of them, a new function is started. For example, when clicking the menu item "open an element", all details of the element is shown in a small window (as shown in Figure 12).

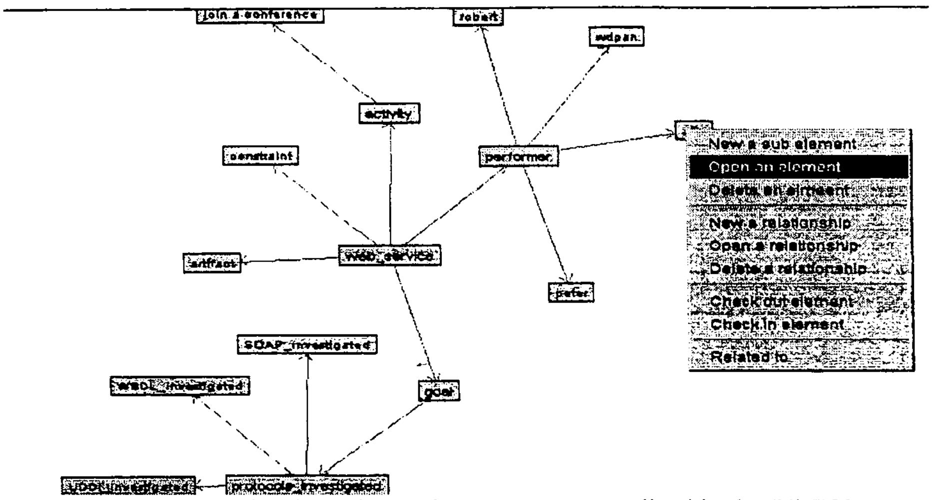

Figure 11: A visual graphic and a popup menu realized in the WbIVC

Particularly, when clicking the menu item "new a process element", a visual input interface is displayed. The participant can input the information of the new element via this interface. After the element is completely input (as shown in Figure 14), the visualization module sends it to the web server via the HTTP-based Communication Module (two methods of the communication module are situated in the visualization module --- a Java applet) so that it is saved in the business process repository.

4. The WbIVC Applied to the Collaborative Business Process Management

The WbIVC is applied to a specific collaborative business process management --- a research project management system (RPMS) in a university environment. The RPMS provides the traditional interfaces (representing process elements and relationships in tables) as well as the visual interfaces (representing process elements and relationships in graphics that consist of nodes and edges). Figure 12 shows a business process that is related to a specific process participant. When the participant wants to review the details of a specific process element, the participant moves the mouse pointer to the rectangle that represents the element, then clicks the right button of the mouse, and then clicks the menu item "open a process element" in the popup menu, and finally a new small window is activated. This small window displays all information of that element.

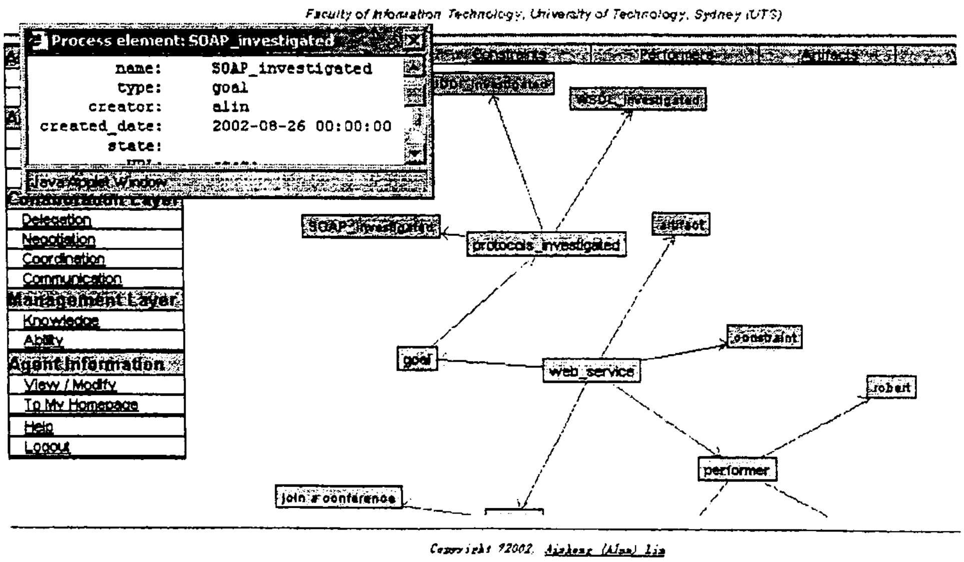

Figure 12: The output interface of the WbIVC in the RPMS

The WbIVC distinctive feature --- accepting a participant's information input and then sending the information to the web server --- is also applied to the RPMS. Figure 13 is an interface that shows a specific participant is creating a new artifact (a Java methods) to a research project. A new editor window is activated and the participant inputs the information in the new window. After the information is completely edited, the participant saves the information to the web server. The WbIVC sends the information to the web server from the web browser and so the information is stored in the web server.

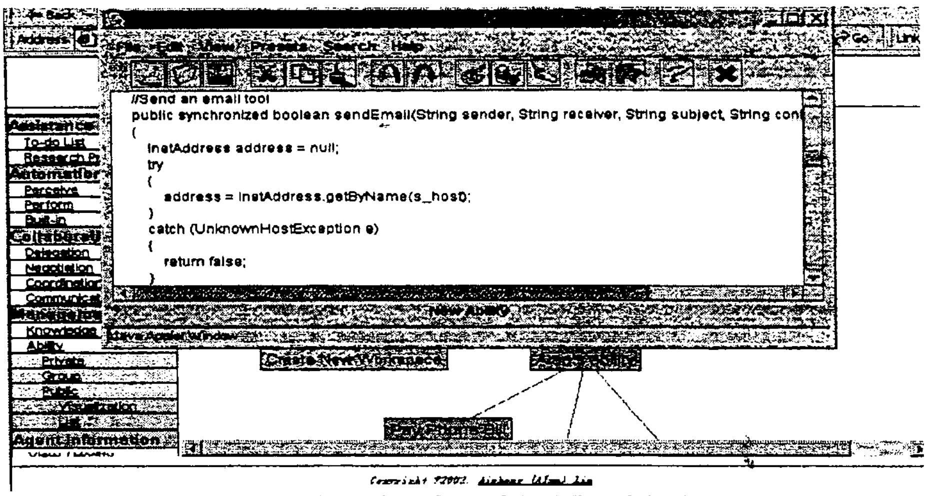

Figure 13: The input interface of the WbIVC in the RPMS

5. Future Work

This paper introduces a web-based interactive visual component (WbIVC) that is employed in our online collaborative business process management system. WbIVC supports collaborative process participants with a visual interface for input/output data as well as manipulating processes. This interface helps process participants in viewing and modifying the process progresses especially for those progresses that are made by other participants. Because of the use of WbIVC, process participants are able to comfortably and conveniently use the RPMS. In the future, we will extend more functions, such as displaying different process element using different shape of graph, into WbIVC for supporting the process participants.

References

[3DAC, 2003] 3D Active Chart. http://www.rinvoice.com/index_en.html

[Card, 1996] Card, S.K. (1996). Visualizing Retrieved Information: A Survey. *IEEE Computer Graphics and Applications, 16(2)*, 63-67.

[*CDPDT, 2002*] F. Corcoran, J. Demaine, M. Picard, L. G. Dicaire, and J. Taylor — "INUIT3D: An Interactive Virtual 3D Web Exhibition", Museums and the Web 2002, Boston, Massachusetts, Apr. 18-20 2002

[DYMOS, 2003] DYMOS: Dynamic Models for Smog Analysis. http://www.first.gmd.de/org/dymos.html

[HEL, 2003] Mao Lin Huang, Peter Eades and Wei Lai. *Online Visualization and Navigation of Global Web Structures. To appear in the International Journal of Software Engineering and Knowledge Engineering,* World Scientific Publishing Company, vol. 13,no. 1, 2003.

[HIT, 1999] Hawryszkiewycz, I.T. "Knowledge Sharing through Workspace Networks". *Proceedings of the Special Interest Group on Computer Personnel Research (SIGCPR99),* April 1999, New Orleans (ISBN 1-58133-063-5), pp. 79-85.

[XML, 2000] Extensible Markup Language (XML) 1.0 (Second Edition). *W3C Recommendation* 6 October 2000. http://www.w3.org/TR/REC-xml