INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 97 TAMPERE, AUGUST 19-21, 1997

THE WORLD WIDE DESIGN LAB: AN ENVIRONMENT FOR DISTRIBUTED COLLABORATIVE DESIGN

Zdenek Zdrahal and John Domingue

Collaborative design, domain ontology, case-based design, knowledge sharing and reuse

1 Introduction

In ever increasing frequency, designers are required to collaborate across large geographical boundaries. This collaboration presents the participants with new challenges. In this paper we describe how we have addressed two of these challenges using Internet technology. The first challenge is *"How can designers discuss complex design artifacts at a distance?"*. Designers' discussions have a complex structure. For example, a designer can refute, justify, or revise a design proposal. These discussions are traditionally supported by sketches and formal diagrams. In this paper we show how our solution allows designers to have the same rich forms of interactions, over the Internet, with the minimum changes to their style of work.

The second challenge we have addressed is *"How can previous design solutions be made reusable between collaborators separated by a large distance?"*. Reusing old solutions is a common design practice. In our experience design institutions maintain an archive of previous designs. Sharing solutions across institutions presents an interpretation problem, as institutions may describe their solutions using different terminology. Organising a shared library also raises challenges. There is a trade-off between reducing the time to find a previous solution and reducing the duplication of design solutions across institutions. Before describing our solutions to these challenges we will now give a brief overview of our approach.

2 Approach

We support design discussion with a tool, described in section 4, which uses a "Chinese wall" metaphor. The metaphor represents design dialogues as multimedia posters. In addition to supporting the metaphor the tool allows designers to present live broadcasts of their interactions with the posters over the Internet.

Design reuse is supported through Case-Based Reasoning (CBR) technology. The application of case-based reasoning to design is not uncommon - several examples are described in [Maher et al., 1995]. Reusing designs (called cases in CBR terminology) from a foreign institution requires a common description language to provide a shared explicit conceptualisation. Within the knowledge modelling community such conceptualisations are called *ontologies.* An ontology defines concepts and relations, for example class hierarchies, within the domain of discourse. The role of ontologies in knowledge sharing and reuse in complex industrial design is described [Borst & Akkermans, 1997]. In section 5 we describe the tool for collaboratively constructing ontologies over the Internet.

In section 6 we describe the organisation of our shared library. Bearing in mind the time/duplication trade-off mentioned above, our dilemma when designing the library centred on the placement of the cases. We decided to distribute the cases across the collaborating institutions. This option results in an efficient use of resources as reused solutions tend to originate locally.

Although in this paper the reusable knowledge we discuss is in the form of delivered solutions, the same principles apply to other types of knowledge. For example, our framework has also

been used for sharing problem solving methods [Motta & Zdrahal, 1996].

For our approach to be accepted by the design community designers must be able to collaborate using their existing hardware and operating system configuration (including their graphical user interface environment). In the next section we describe how we satisfied this constraint using Web server and Java technology.

3 Architecture

The architecture has two main components a central WW Design Lab Server and a WWDL client. The central server is composed of three layers: an Internet infrastructure layer, a knowledge tool layer and a design layer. The Internet infrastructure layer is responsible for interfacing to the Web and email and allowing user interface actions to be broadcast. This layer is composed of a customised Web server, an email handler and a broadcaster. The customised Web server is based on LispWeb [Riva & Ramoni, 1996] a specialised HTTP server written in Common Lisp. In addition to implementing the standard HTTP protocol, the LispWeb server offers a library of high-level Lisp functions to dynamically generate HTML pages, a facility for dynamically creating image maps, and a server-to-server communication method.

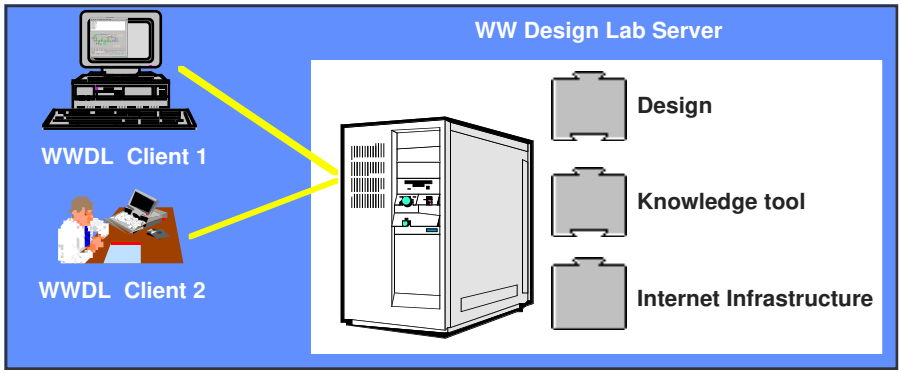

Figure 1. The general architecture of the World Wide Design Laboratory.

The email handler parses incoming email and sends on the decomposed parts to interested knowledge tools. All WWDL clients have broadcast and receive modes. All the interface actions of broadcasting clients are copied to receiving clients. When broadcasting or receiving a WWDL client maintains an open TCP/IP stream to the server. The broadcaster is responsible for distributing the incoming broadcast data to the receivers.

The knowledge support layer enables reusable design ontologies to be constructed and partial solutions to be stored and retrieved. The layer is based on OCML an operational knowledge modelling language [Motta, 1995] and a Case Based Reasoning [Zdrahal & Motta, 1996] tool.

The design layer currently contains a case library. Future work will expand the library of design problem solving methods [Motta & Zdrahal. 1996]

A designer interacts with WWDL using one of our WWDL design clients. There is a client for each of the knowledge support tools and a client for gaining an overview of a design.

4 Tadzebao

Tadzebao is a Chinese word (which literally means “Big Character Poster”) used in Italy to describe the political debating process used during the Cultural Revolution. During the Cultural Revolution a political argument or ideology would be expressed through the placement of a poster. Rebuttals to or comments on the initial argument would be expressed by additional posters on top of or around the original poster.

WWDL uses an extension of this metaphor for guiding designers around ongoing design dialogues. Within Tadzebao, dialogues are centred around a *notepad* which contains a series of pages. Each notepad page can contain a mixture of text, GIF images, hand drawn sketches and design knowledge encoded in one of the knowledge tools. Dialogues can be browsed by clicking on the bottom right corner of a notepad.

Figure 2 below shows six notepads each containing a design dialogue. We shall now briefly describe the top page of each of these notepads in turn.

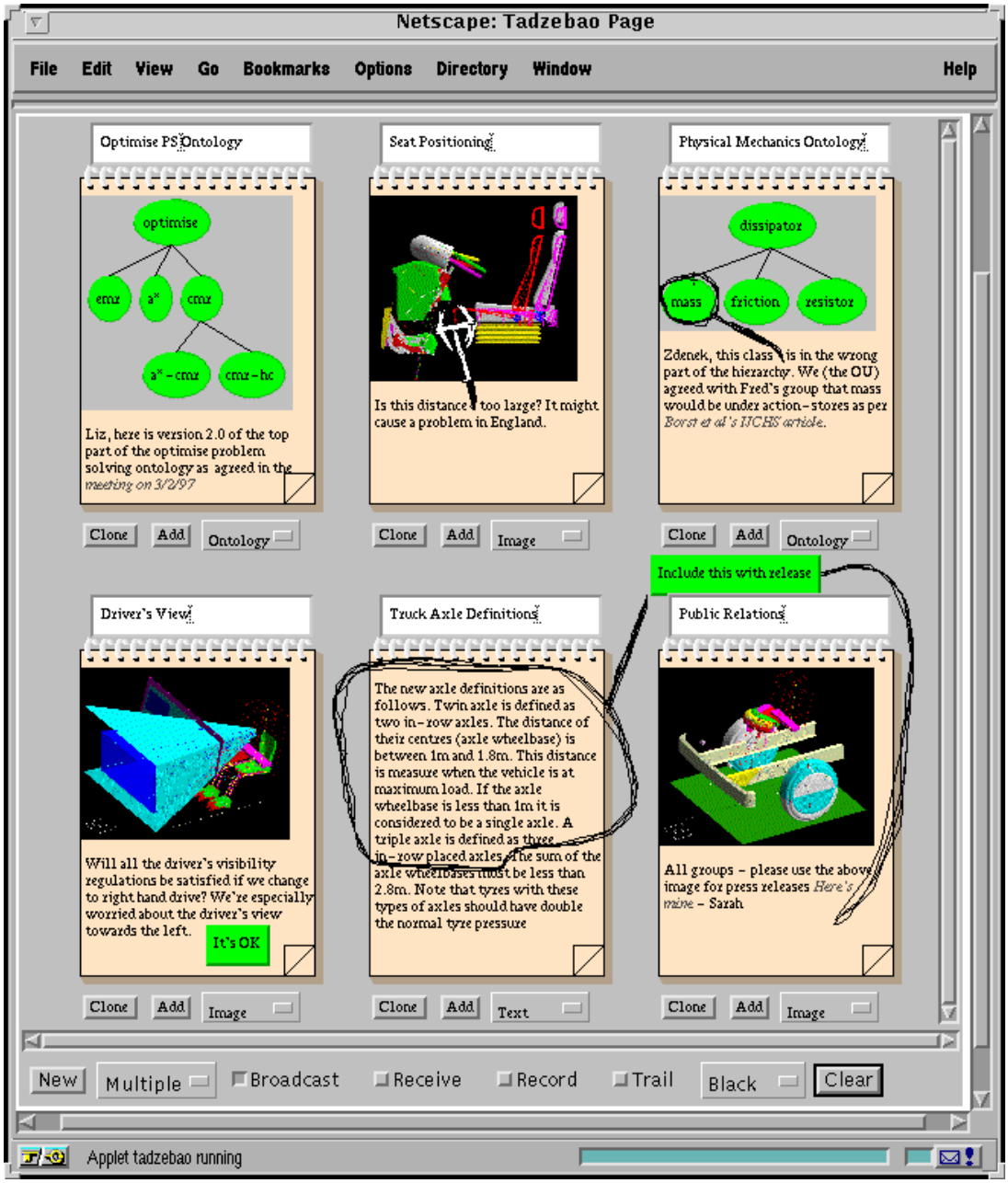

Figure 2. A screen snapshot of Tadzebao in use.

Optimise PS Ontology - this page shows the top portion of a new design ontology and some textual comments. The designer added the ontology by clicking on the "Add" button and choosing an ontology from a displayed list. The "Add" button adds one of: a text area; GIF image; or ontology depending on the current choice (currently showing "Ontology"). The text comments were added using a simple text editor provided within the environment. The text displayed in grey italics ("meeting on 3/2/97") is a hypertext link to the relevant page within the notepad.

Seat Positioning - this page contains a query on the distance between the driver's seat and the pedals within a truck cabin. The GIF image was added by specifying it's Web address. The white (and black) sketched arrow specifying the distance were drawn using the mouse after clicking on the "Trail" button in the bottom control panel.

Physical Mechanical Ontology - this page shows a designer disagreeing with the structure of an ontology. The text displayed in grey italics ("Borst et al's IJCHS article") is a hypertext link to

the article. Clicking on an ontology allows it to be edited within our ontology editor. The screen snapshot shown in figure 3 below was created by clicking on the ontology shown in this page.

Driver's View - the designer has added a reply to the visibility query by cloning the original page, using the "Clone" button, and then sticking on a label (like a "Post-It™" sticky note) with the phrase "It's OK". It is possible to clone an entire notepad if a designer wishes to view two pages at once.

Truck Axle Definition - this plain text page was emailed to Tadzebao, using an address known to the email handler, with the subject line "Truck Axle Definitions" - the name of the notepad. The body of the message, describing new definitions of truck twin and triple axles, was then added to the top of the notepad. It should be noted that Tadzebao can also handle GIF image attachments to email messages.

Public Relations - this text is a note delineating the use of a particular image for a press release. Clicking on an image creates a new window showing a full size version. The sketched circle around this page and the *Truck Axle Definition* page, and label "include this with release" are part of a live broadcast. Selecting the "Broadcast" button in the bottom control panel broadcasts all interface actions (including all sketching, page turning and adding of text, GIF images or ontologies) to *receivers,* Tadzebao's with the "Receive" button selected, in real time. The designer is using this to tell the public relations officer (Sarah) that it would be beneficial to have part of the axle definitions included with the press release. A broadcast can be recorded, if required, by selecting the "Record" button. Recordings are stored in a "movie" database and can be retrieved and replayed when convenient.

5 Web Ontology Browser

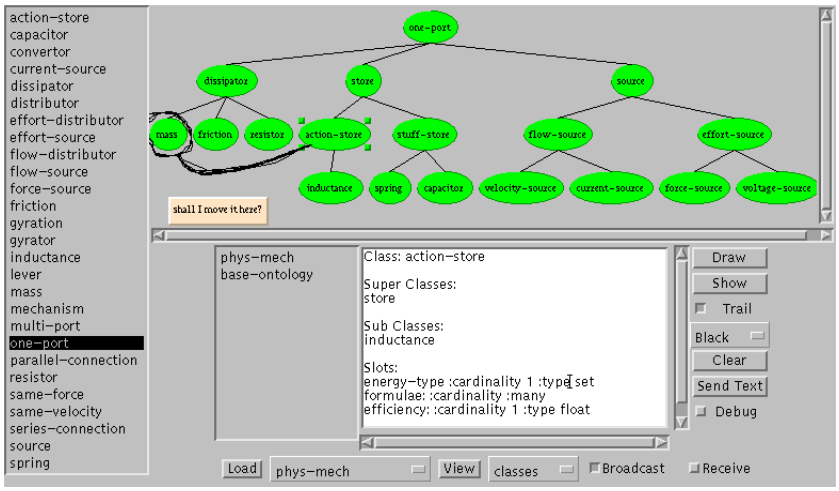

Figure 3. A screen snapshot of the Web Ontology editor.

Figure 3 above shows a screen snapshot of the Web ontology editor. The list on the left shows all the classes for the physical mechanical ontology (taken from [Borst et al. 1997] p. 375). This list can also be used to view the functions and relations within an ontology - by changing the button currently showing "classes" in the bottom control panel and then hitting the "View" button.

Within the list, the class "one-port" has been selected and drawn using the "Draw" button on the right side of the screen. Within the graphically displayed class hierarchy the node "action-store" has been selected and inspected using the "Show" button. This displays the information shown in the white central window. As with Tadzebao the Web Ontology editor can be put into broadcast mode. Figure 3 shows a user discussing with a colleague (whose Web ontology editor is in receive mode) the problem highlighted in the *Physical Mechanics Ontology* notepad (see figure 2). The user's sketches are displayed simultaneously on the receiving client.

6 Case-Based Design Tool

The generic computational model of case-based reasoning consists of a *case-library* and a *reasoning engine*. The engine performs two major tasks: case retrieval, and adaptation. The case retrieval task involves finding the case with the best match to the problem specification. This case becomes the basis for the solution. In the adaptation step the case is evaluated and potential discrepancies are fixed.

Care needs to be taken when instantiating the above generic computation model with a distributed case library. Specifically, there is a need to reduce the number of Internet accesses. A high number of Internet accesses could make the tool too unresponsive to be of practical use. The rest of this section focuses on our solution to this problem.

Let us consider the retrieval algorithm in detail. For each case we calculate a *similarity measure* between it and the current requirements. Then case with the highest score is downloaded. As we have opted for a distributed organisation of our case library we need to ensure that we do not invoke an Internet access for each similarity measure calculation, as this would notably degrade the performance of the tool.

Our solution uses the agent based architecture shown in figure 4. The retrieval task has been delegated to three collaborating software agents: a book-keeping agent (BKA), a case-matching agent (CMA) and a request manager agent (RMA). The book-keeping agent maintains a *case descriptor library (CDL)* - a minimum collection of case data required for identifying the best case. Finding the best cases and presenting them to the designers is responsibility of the clients' request manager agents and the case-matching agent.

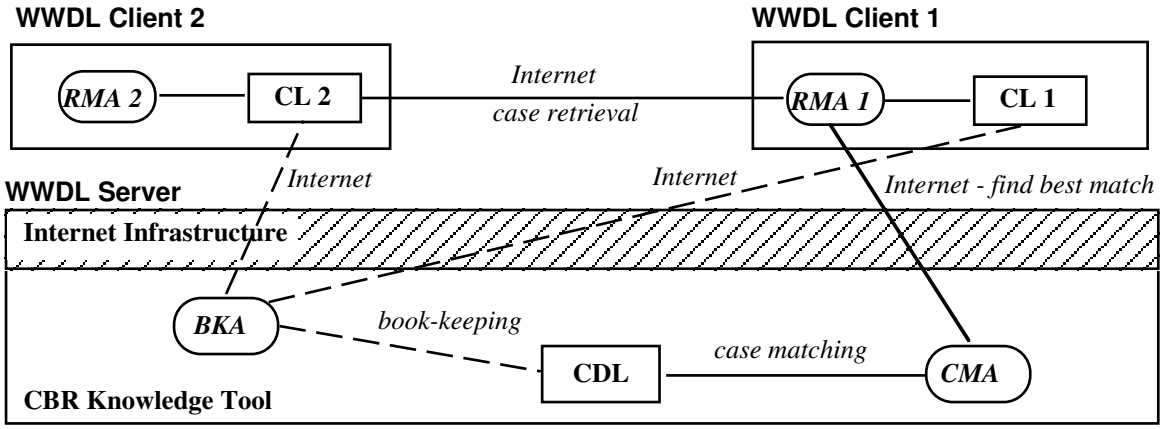

Figure 4. The agent based architecture of the case-based design tool. The abbreviations in the figure are: BKA - book-keeping agent; CMA - case-matching agent; RMA - request manager agent, CL - case library, and CDL - case descriptor library.

Let's consider a small example using figure 4. Imagine that Client 1 initiates a case retrieval. Client 1's RMA communicates this request to the case-matching agent. From the data in the case descriptor library (CDL) the CMA selects the best case, which in this example resides in case library 2 (CL 2). The CMA returns the Internet address of CL 2 and the identifier of the best case to client 1's RMA. Client 1's RMA then downloads the best case from CL 2. A potentially large number of Internet queries is reduced to 4 (RMA 1 -> CMA, CMA -> RMA 1, RMA 1 -> CL 2, CL 2 -> RMA 1). This number is independent on the size of the distributed case library.

The book-keeping agent periodically updates the CDL by inspecting remote case libraries (CL 1 and CL 2 in figure 4). The update interval is set to correspond to the collaborating design groups work schedule. If a designer produces new result and immediately wants to make it available to the remote partners he/she can force an exceptional CDL update. In this instance the designer's RMA sends an update request to the book-keeping agent. This update strategy ensures that at any time the CDL contains up-to-date information. Designers can also send update requests to the book-keeping agent by e-mail.

7 Conclusions

In this paper we have described a set of tools which allow designers to share design knowledge across space and time. Using Tadzebao designers can take part in synchronous and asynchronous design dialogues via a rich medium. Future versions of Tadzebao will incorporate audio annotations. The Web Ontology editor allows designers to collaboratively construct an explicit shared conceptualisation which facilitates design reuse. The CBR tool allows designers to reuse previous results across large distances.

The CBR tool is currently being tested using a case library distributed over: Iowa University in the US, the Czech Technical University in Prague, the Technical University at Kosice and at the Knowledge Media Institute, the Open University in the UK. Our architecture means that the number of Internet accesses for a case retrieval is a small constant, independent of the both the number of collaborating partners and the number of cases within the distributed case library. We are also using the Web server/Java client architecture in a number of other contexts (e.g. software maintenance [Domingue & Mulholland, 1997]).

A tool similar to Tadzebao was found to be of great use in the large collaborative engineering project Madefast [Cutkosky et al., 1996]. Unlike Madefast, all WWDL tools are platform independent. Engineering design is a complex time consuming task, our contribution is to enable designers to easily share the fruits of their labour.

References:

Borst, P and Akkermans, H., "Engineering ontologies", Int. Journal of Human - Computer Studies, Vol. 46, No.2/3, 1997, pp. 365-406.

Cutkosky, M. R., Tenenbaum, J. M. and Glicksman, J., "Madefast: Collaborative Engineering over the Internet", Communications of the ACM, Vol. 39, No. 9, 1996, pp. 78-87.

Domingue, J. and Mulholland, P., "Fostering Debugging Communities on the Web", To appear in Communications of the ACM, April 1997

Maher M.L., Balachandran M.B. and Zhang D.M., "Case-Based Reasoning in Design", Lawrence Erlbaum Associates, Publishers, Mahwah, New Jersey, 1995

Motta E., "KBS Modelling in OCML", Modelling Languages for KBS, VU Amsterdam, 1995

Motta E., and Zdrahal Z., "Parametric Design Problem Solving", Proceedings of the 10th KAW, (B.Gaines & M.Musen eds.), Banff, Canada, 1996

Riva, A. and Ramoni, M., "LispWeb: a Specialized HTTP server for Distributed AI Applications", Computer Networks and ISDN Systems, 1996, Vol. 28, No. 7-11. pp. 953-961

Zdrahal, Z. and Motta, E., "Case-Based Problem Solving Methods for Parametric Design Tasks", In Advances in Case-Based Reasoning (I.Smith & B.Faltings eds.), Springer, 1996, pp. 473-486.

Zdenek Zdrahal
The Knowledge Media Institute,
The Open University,
Walton Hall,
Milton Keynes, MK7 6AA
United Kingdom
Tel: +44 1908 654512
Fax: +44 1908 653169
Email: Z.Zdrahal@open.ac.uk