C3W: Clipping, Connecting and Cloning for the Web

Jun Fujima[*]
Meme Media Laboratory
Hokkaido University
N13 W8, Sapporo 060-8628, Japan
fujima@meme.hokudai.ac.jp

Aran Lunzer
Natural Sciences ICT Competence Centre
University of Copenhagen
2100 Copenhagen Ø, Denmark
alunzer@nik.ku.dk

Kasper Hornbæk
Natural Sciences ICT Competence Centre
University of Copenhagen
2100 Copenhagen Ø, Denmark
khornbaek@nik.ku.dk

Yuzuru Tanaka
Meme Media Laboratory
Hokkaido University
N13 W8, Sapporo 060-8628, Japan
tanaka@meme.hokudai.ac.jp

ABSTRACT

Many of today's Web applications support just simple trial-and-error retrievals: supply one set of parameters, obtain one set of results. For a user who wants to examine a number of alternative retrievals, this form of interaction is inconvenient and frustrating. It can be hard work to keep finding and adjusting the parameter-specification widgets buried in a Web page, and to remember or record each result set. Moreover, when using diverse Web applications in combination – transferring result data from one into the parameters for another – the lack of an easy way to automate that transfer merely increases the frustration. Our solution is to integrate techniques for each of three key activities: clipping elements from Web pages to wrap an application; connecting wrapped applications using spreadsheet-like formulas; and cloning the interface elements so that several sets of parameters and results may be handled in parallel. We describe a prototype that implements this solution, showing how it enables rapid and flexible exploration of the resources accessible through user-chosen combinations of Web applications. Our aim in this work is to contribute to research on making optimal use of the wealth of information on the Web, by providing interaction techniques that address very practical needs.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Input devices and strategies, Interaction styles*; H.5.4 [**Information Interfaces and Presentation**]: Hypertext/Hypermedia

General Terms

Design, Human Factors

Keywords

Web navigation, Web application linkage, IntelligentPad, Subjunctive interfaces

1. THE MECHANISMS OF C3W

Suppose an investor in Japan wants a way to look up the stock price of US-quoted companies, but to see them in Japanese yen rather than dollars. The investor knows that the CNN Money site offers a stock-price lookup in dollars (money.cnn.com/markets/), while Yahoo! has a currency conversion page that can certainly deal with dollars and yen (finance.yahoo.com/m3?u). Furthermore, suppose the investor wants to see a number of different companies' prices at the same time. We describe how C3W's mechanisms allow this investor to build a new tool addressing these wishes.

1.1 Clipping

C3W has been developed as an extension to the IntelligentPad system [3]. IntelligentPad represents information and processing components as 2D visual objects called pads. Our support for clipping directly from Web pages is implemented by cooperation between an enhanced Web-browsing pad and a component that we call a DerivationPad.

The browser supports the extraction of arbitrary elements from within encountered Web pages, as visual cells that function as portals onto those pages. With drag-and-drop mouse operations the user extracts these cells and arranges them on a DerivationPad, where each cell retains its role as a region for either parameter input or result display. Cells are typically extracted from a navigation that spans multiple Web pages, and carry detailed source information that allows the DerivationPad to determine their mutual relationships. Thus when a user supplies a new value for an input cell, the appropriate navigation steps are replayed to deliver results to any associated result cells.

As the user performs a navigation using our browser, a navigation path is recorded. The path is a sequence of steps representing encountered documents. A step consists of the URI of the document, a 'predicate' specifying any conditions that the user has specified on that document, and 'targetElement' specifying the interaction that took the browser to the next document (if any). The predicate conditions correspond to values specified in elements such as INPUT or SELECT; the targetElement is typically an A (anchor) or FORM. For these attributes we use HTML-path expressions [1] as the way to identify arbitrary portions of an HTML document. An HTML-Path is a specialisation of an XPath expression, with functions specialised for HTML.
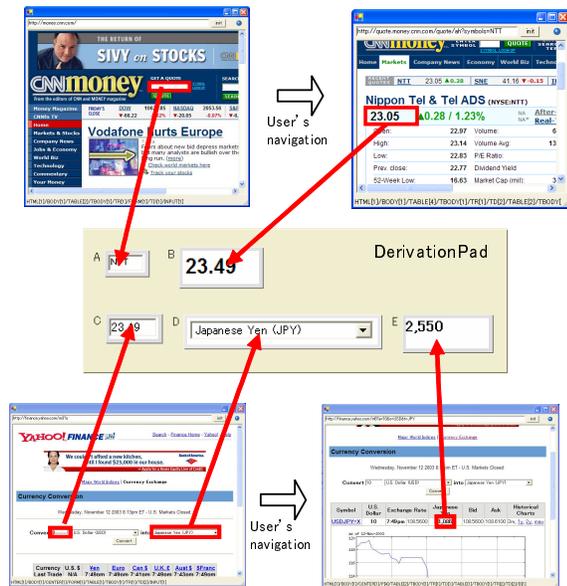
Figure 1: Clipping from Yahoo! and CNN Money

Figure 1 shows the user clipping the desired input and result elements from the CNN and Yahoo! sites. From CNN the user clips the input field used for specifying a company code, then enters a sample code and requests its price, then clips the price display from within the result page. Similarly, the user clips the currency-amount input field and the destination-currency menu from the Yahoo! conversion page, enters a sample amount and presses 'Convert' , then clips the conversion result.

1.2 Connecting

C3W's second C refers to the ability to Connect applications with the help of inter-cell formulas, so that derivations that are normally driven by user input can instead be driven by the results of other processing.

Figure 2: Connecting two applications, using a formula

In Figure 2, the user has already taken the final step necessary in connecting the applications together: establishing that the conversion-amount cell (C), rather than its previous status as a user input field, is to take its value from cell B. This is done by adding the simple formula '=B' to cell C. Now whenever B's value changes, as the result of a new stock-price lookup, C is automatically updated and a currency conversion is triggered. If the user wanted instead to convert the price of 100 shares in the specified company, this could be done by making C's formula '=B*100'.

This DerivationPad also works as a reusable application, accepting a stock code in cell A and producing a converted price in E. It could be configured to re-evaluate automatically every few minutes to reflect the latest stock price and conversion rate.

1.3 Cloning

The idea of cloning comes from the motivating theme of our work on Subjunctive Interfaces [2]: that users often need to compare alternative scenarios. An application with a subjunctive interface allows its user to set up many scenarios, typically involving different values for the application's input parameters; to see all those scenarios in parallel; and to adjust scenarios in parallel.

In an IntelligentPad environment, multiple scenarios can be supported by cloning the interface pads that handle an application's inputs and results, and orchestrating their connections to the 'parent' pads that perform the processing. Thus at one moment a processing pad will be connected to the interface pads for scenario 1, and will perform any necessary processing for that scenario; once this has finished, it will be connected instead to the interface for scenario 2; and so on. As far as the user is concerned, all the scenarios appear to operate independently. This makes them useful for side-by-side comparisons.
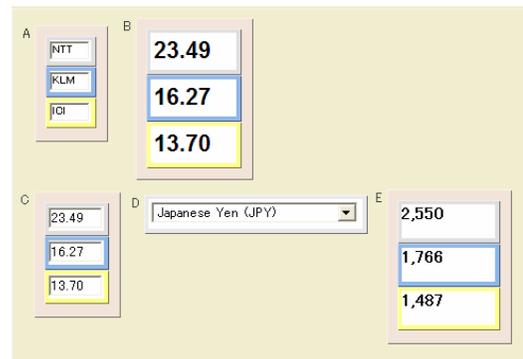

Figure 3: Cloning to support three retrievals in parallel

Figure 3 shows several companies' stock prices in parallel. When the user types a new stock code into one of the instances of cell A, the results for that stock will appear in the corresponding instance of each of the derived cells, including the final result E.

2. CONCLUSION

We have introduced C3W, a prototype that supports users in accessing diverse resources available through Web applications. Its support centres on information cells that the users visually clip from Web pages, connect with formulas as the way to link applications, and clone to allow parallel viewing and control of alternative retrievals. The working prototype has allowed us to build many examples that demonstrate the promise of our approach; we are now working to increase the generality of its three core techniques, with the aim of creating an interface that significantly enhances users' opportunities to benefit from the world's Web resources.

3. REFERENCES

[1] K. Ito and Y. Tanaka. A visual environment for dynamic web application composition. In *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 184–193. ACM Press, 2003.

[2] A. Lunzer and K. Hornbæk. Usability studies on a visualisation for parallel display and control of alternative scenarios. 2004. To appear.

[3] Y. Tanaka. *Meme Media and Meme Market Architectures: Knowledge Media for Editing, Distributing, and Managing Intellectual Resources*. IEEE Press, 2003.