Paper presented to XII Congress on PCP, Seattle, USA, July 8-12 1997

Using Grids to Develop Expert Systems through the World Wide Web

Mildred L. G. Shaw and Brian R. Gaines
Knowledge Science Institute
University of Calgary
Alberta, Canada T2N 1N4
{mildred, gaines}@cpsc.ucalgary.ca

Abstract: WebGrid-II is a knowledge acquisition and inference server on the World Wide Web that uses an extended repertory grid system for knowledge acquisition, inductive inference for knowledge modeling, and an integrated expert system shell for inference. This article describes WebGrid-II, and illustrates its use in eliciting and modeling knowledge. It shows how personal construct psychology provides the foundations for knowledge-based systems development and application.

1 Introduction

The development of knowledge-based systems involves knowledge acquisition from a diversity of sources often geographically distributed. The sources include books, papers, manuals, videos of expert performance, transcripts of protocols and interviews, and human and computer interaction with experts. Expert time is usually a scarce resource and experts are often only accessible at different sites, particularly in international projects. Knowledge acquisition methodologies and tools have developed to take account of these issues by using hypermedia systems to manage a large volume of heterogeneous data, interactive graphic interfaces to present knowledge models in a form understandable to experts, rapid prototyping systems to test the models in operation, and model comparison systems to draw attention to anomalous variations between experts.

However, existing knowledge acquisition tools are largely based on personal computers and graphic workstations, and their use in a distributed community involves moving software and often computers from site to site. The process of building and testing knowledge models across a distributed community would be greatly expedited if wide-area networks could be used to coordinate the activities at different sites. The initial objective of the work reported in this paper has been to use the World Wide Web to support distributed knowledge acquisition by porting existing knowledge acquisition tools to operate through the web. A further objective has been to use the web on a continuing basis to support distributed knowledge management in which the acquisition, representation and application of knowledge become an integral part of the activities of an organization.

Repertory grids based on personal construct psychology (Kelly, 1955) have been used for knowledge acquisition since the early years of knowledge-based system development (Shaw and Gaines, 1983; Boose, 1984) and have been refined over the years to support increasingly complex knowledge structures (Boose, Bradshaw, Koszarek and Shema, 1993; Ford, Bradshaw, Adams-Webber and Agnew, 1993; Gaines and Shaw, 1993a). Since grid elicitation tools are used directly by experts it would be very useful to have them accessible through any personal computer or workstation with access to the web. This is feasible because the primary data input format is through rating scales, and this can be done effectively by using popup menus which are available through the HTML graphic user interface, and is the basis of WebGrid (Shaw and Gaines, 1995). WebGrid-II extends WebGrid with new data types and analysis techniques to support expert system development.

The next section gives an overview of the web architecture and protocols, and following sections illustrate how knowledge acquisition systems have been ported to the web to operate in a distributed client-server environment.

2 The Architecture of the World Wide Web

The World Wide Web was conceived by Berners-Lee in March 1989 as a "hypertext project" to organize documents at CERN (Berners-Lee and Cailliau, 1990). The design involved a simple hypertext markup language that authors could enter through a word processor, distributed servers running on machines anywhere on the network, and access through any terminal. In March 1993 the web was still being presented as primarily a hypermedia retrieval system (Berners-Lee, 1993), but in November that year a development took place that radically changed the nature of the web. Andreessen (1993) issued NCSA Mosaic version 2 using tags to encode definitions of Motif widgets (popup menus, check boxes, text entry boxes, etc) embedded within a hypermedia document encoded in the hypertext markup language (HTML), and allowed the state of those widgets within the client to be transmitted to the server. With the addition of such *forms* providing basic graphic user interface capabilities, the web protocols became the basis of general interactive, distributed, client-server information systems.

Figure 1 shows the client-server architecture of the web. A client accesses servers on the Internet using various protocols. It communicates with various helper applications that extend its functionality. When it accesses a web server using the hypertext transfer protocol (HTTP) protocol that server can also access various helper applications through server gateways.

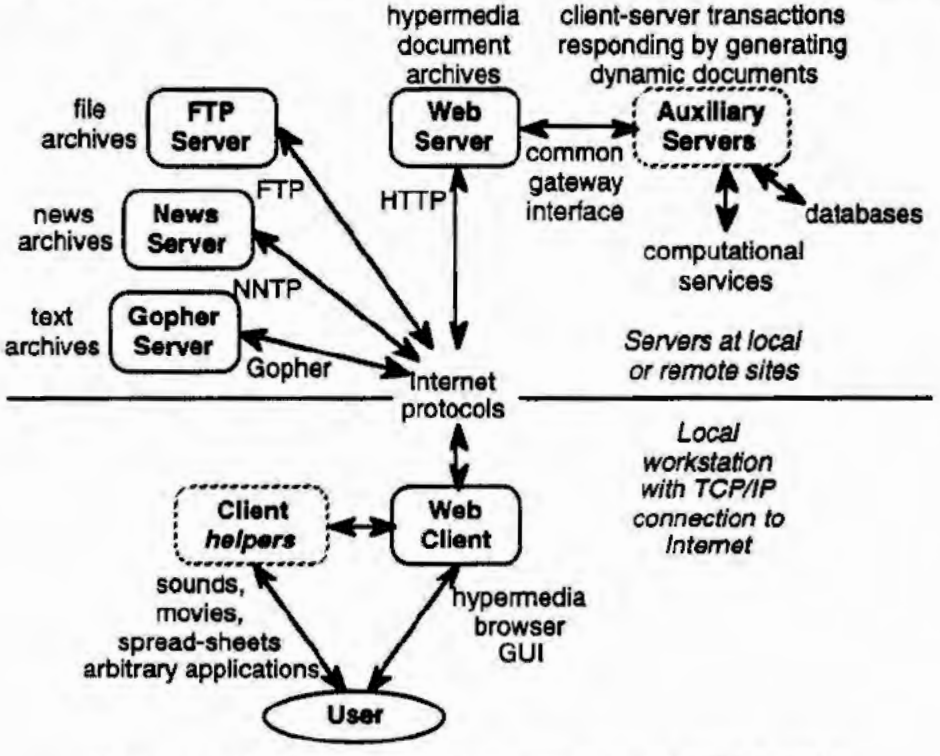

Figure 1 Client-server architecture of World Wide Web

The following sections exemplify the implementation of major knowledge acquisition tools as client helpers and auxiliary servers operating through the web.

3 WebGrid-II: Repertory Grid Client-Server Implementation

WebGrid was developed in 1994 to offer repertory grid elicitation and analysis through the World Wide Web. The auxiliary server in Figure 1 is our RepGrid/KSS0 program (CPCS, 1993) which operates on an Apple Macintosh and offers a wide range of elicitation and analysis capabilities for single and multiple repertory grids and has been widely used in expert system development (Gaines and Shaw, 1993b; Shaw and Gaines, 1993; Shaw, Gaines and Linster, 1994). Offering access to RepGrid through the web made much of its functionality available to users throughout the world through standard web browsers operating on any computer.

WebGrid was intended primarily to introduce repertory grid techniques to new users and focused on basic capabilities. However, it rapidly came into widespread use for major research projects and interest grew in having a full range of knowledge acquisition tools available on the web. WebGrid-II extends WebGrid to provide integration with multimedia, additional data types in the grid, entailment analysis to induce rules from grid data, and an expert system inference engine to allow the rules to be tested on new cases as part of the elicitation process.

3.1 Elicitation

Figure 2 shows the initial screen of WebGrid-II. The HTML form requests the usual data required to initiate grid elicitation: user name; domain and context; terms for elements and constructs; default rating scale; data types allowed; and a list of initial elements. It also allows the subsequent screens to be customized with an HTML specification of a header and trailer—this capability to include links to multimedia web data is also used to allow annotation, text and pictures, to be attached to elements.

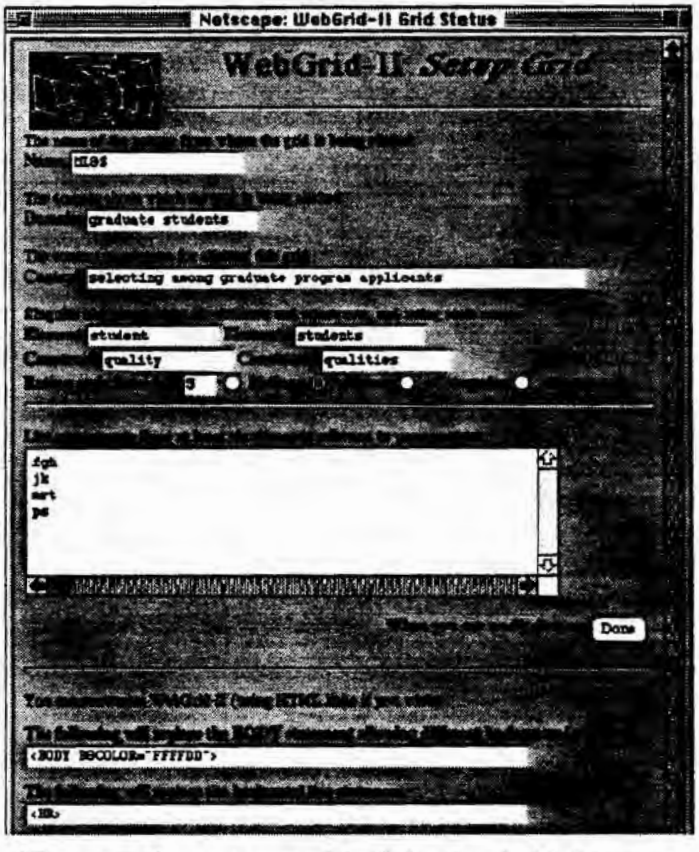

Figure 2 Repertory grid elicitation initial screen

The problem domain is that of "selecting among graduate program applicants" and the intention of the user is to develop a set of rules defining the criteria for selection. The system will incorporate the text typed in by the user into the later dialog, and the user has stated that the elements in the grid are to be called "students" and the constructs "qualities." The user has set the types of constructs to be "Ratings+Names"—the "Ratings" option elicits a standard repertory grid and the "+Names" option indicates that the user wants to use more advanced facilities for knowledge modeling. She has left the default rating scale to be 1-5 and has entered the initials of 4 students typifying the range of selection decisions normally made.

When the user clicks on the "Done" button at the bottom, the browser transmits the data entered to the remote server which passes it through its common gateway interface to a specialist knowledge acquisition auxiliary server. The server processes the data and generates an HTML document that it returns to the browser resulting in the screen shown in Figure 3 eliciting a construct from a triad of elements.

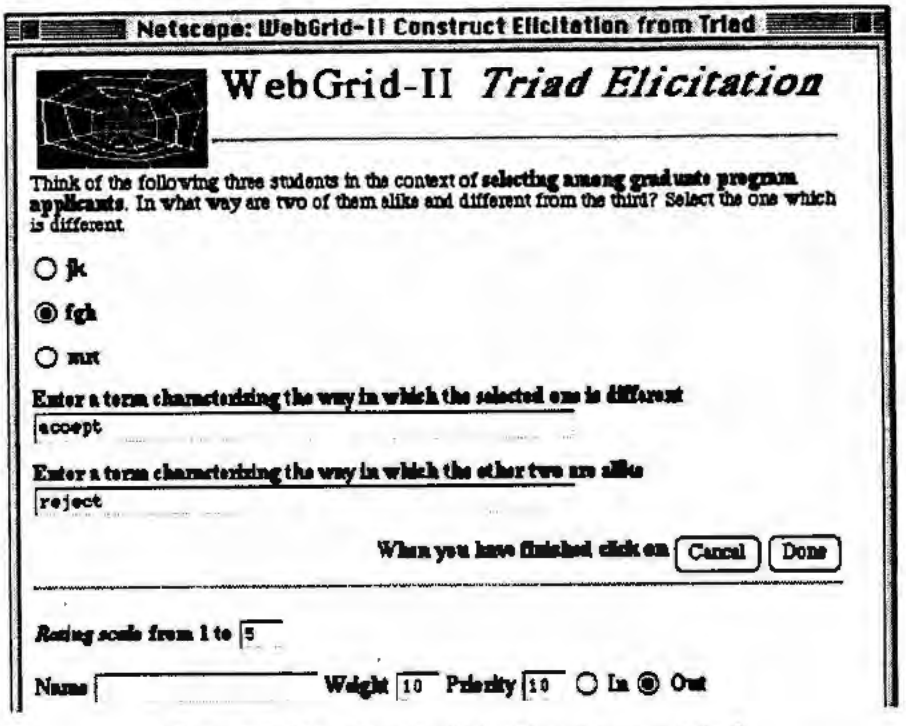

Figure 3 Construct elicitation from a triad

The user clicks on a radio button to select an element which she construes as different from the other two and enters terms characterizing the construct. The lower part of the screen is generated because advanced features were selected through the "+Names" option. It allows the user to vary the rating scale for an individual construct, to name the construct, to give it a weight in clustering, a priority in asking questions in an expert system application, and to specify that the construct is an input for decision making purposes or an output to be anticipated. Since the construct entered is the main decision to be made, to accept or reject a student, and the user wants the system to anticipate the correct decision from the way the student is construed she has made it an output.

When the user clicks on "Done" the server generates the screen shown on the left of Figure 4 which places a popup menu rating scale alongside each element enabling the user to rate each one along the new construct as shown on the right. She is also able to change the terms used if they seem inappropriate in the context of all the elements, and to change the ratings of the already entered elements if appropriate.

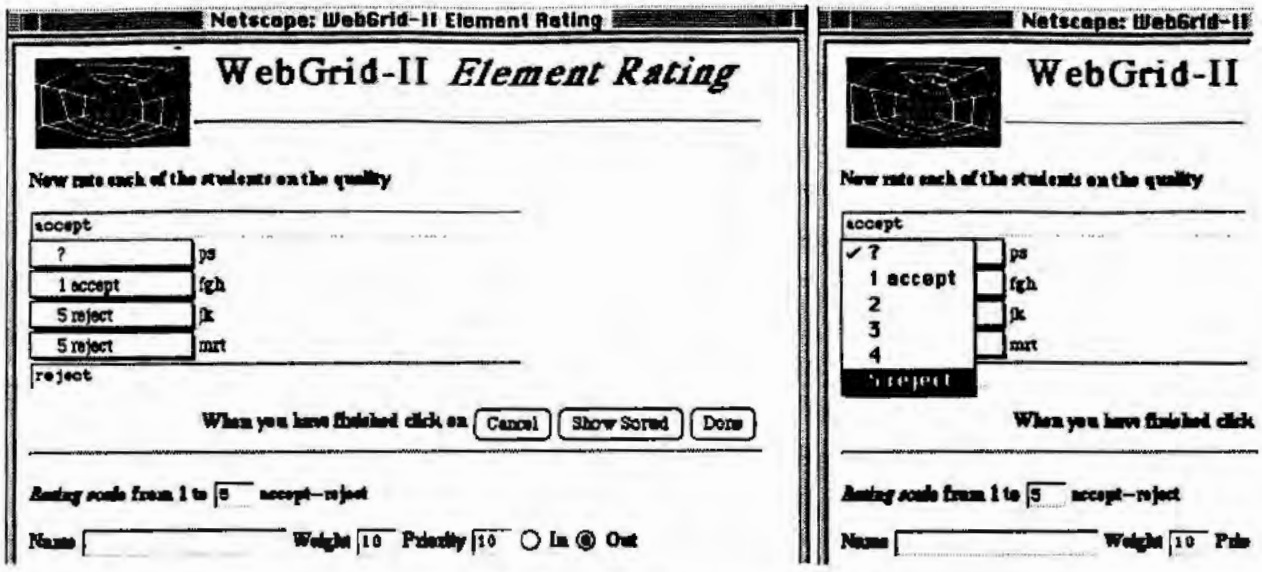

Figure 4 Rating elements on constructs

Clicking on the "Done" button in Figure 4 sends the ratings back to the server which generates the status screen shown in Figure 5.

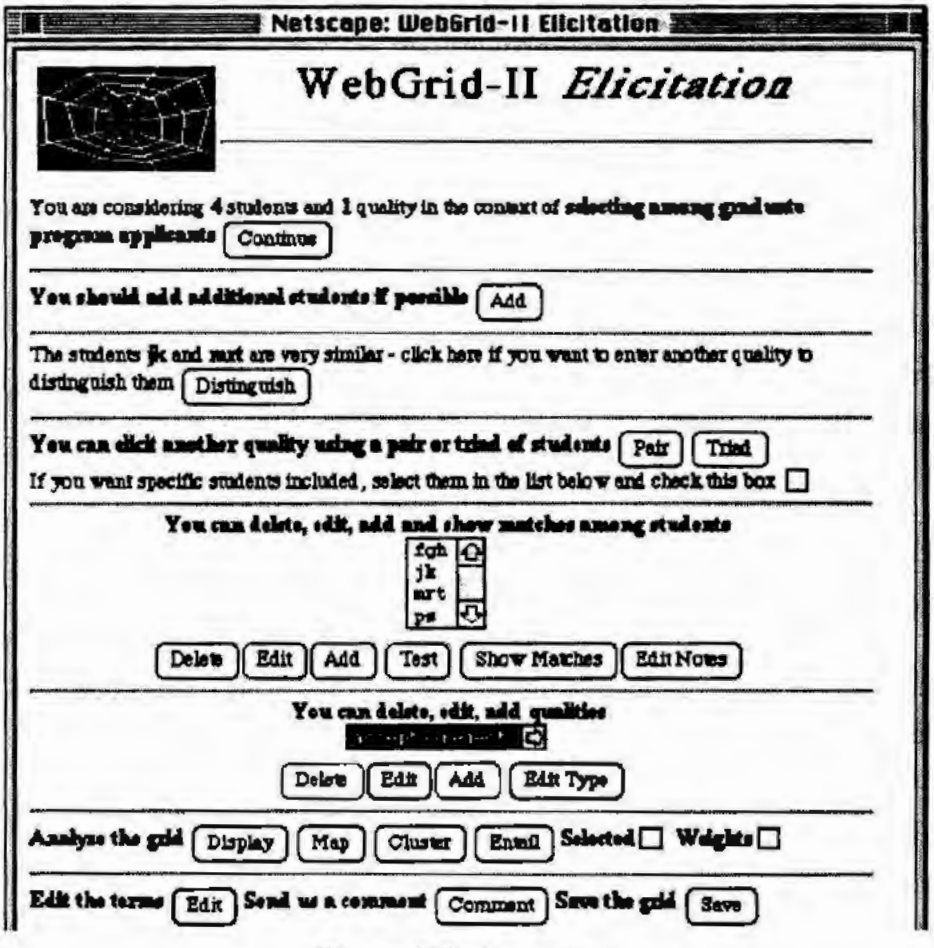

Figure 5 Status screen

This shows the elements and constructs entered, allowing them to be selected for deletion, editing and so on. It also offers various suggestions as how to continue the elicitation based on the data entered so far, facilities for analysis, saving the grid, and so on. The user decides to develop another two constructs from triads of elements and enters "high gpa—inadequate gpa" and "good references—inadequate references" since she knows that these are critical to acceptance.

3.2 Entailment Analysis through Rule Induction

The status screen now shows 3 constructs and the user clicks on the "Entail" button near the bottom of Figure 5 hoping the data is sufficient to generate a model of the way in which the constructs entered entail the "accept—reject" decision. The system develops a model using the Induct algorithm (Gaines, 1989, 1991) for inductive derivation of entailments and generates the screen shown in Figure 6.

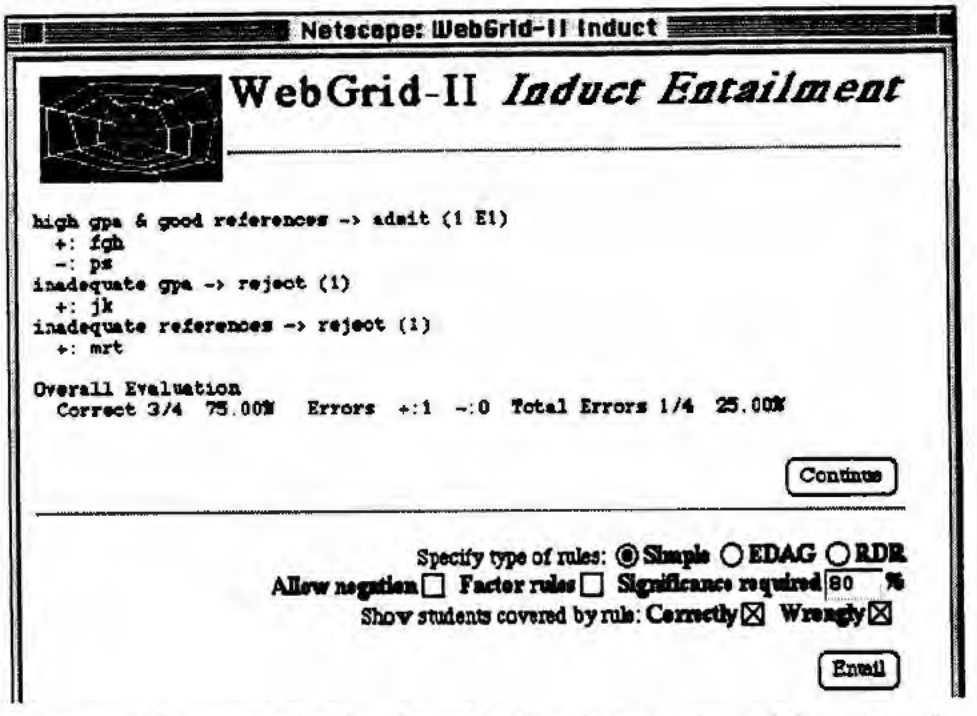

Figure 6 Rules entailed by the grid after 4 elements and 3 constructs

She is disappointed to receive the entailments shown which look sensible but work for only 3 cases in 4. The output consists of a set of rules defining the anticipatory entailments and each rule is followed by the positive elements where the correct decision is anticipated and the negative ones where that anticipated is incorrect. On thinking about it she realizes that the student "ps" has a high gpa and good references but was rejected because there was no-one willing to supervise him. She returns to the status screen, adds a construct "willing supervisor—no supervisor" and clicks on "Entail" again. This time the rules entailed by the grid returned as shown in Figure 7 correctly anticipate the decisions made for all 4 cases.

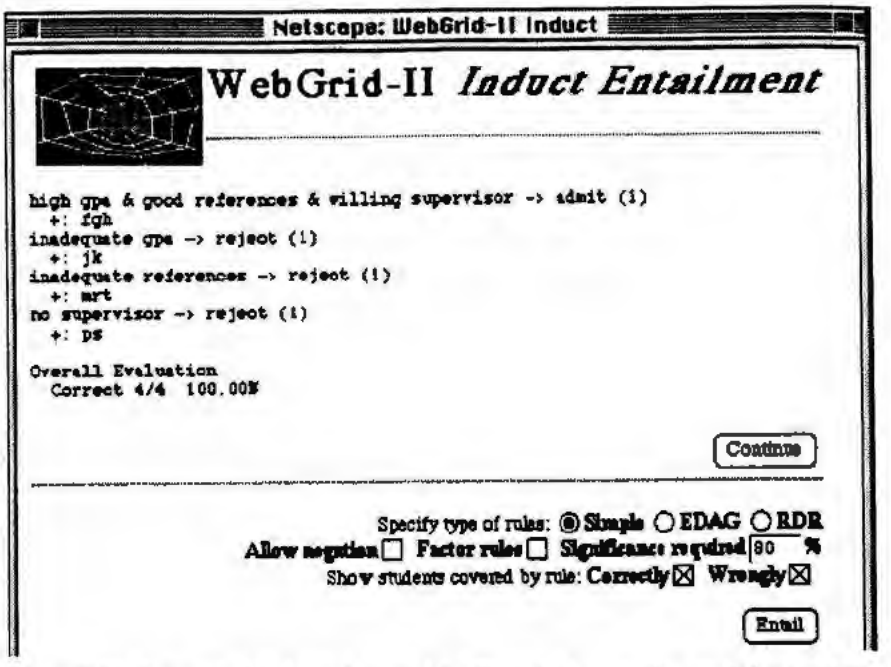

Figure 7 Rules entailed by the grid after 4 elements and 4 constructs

The user knows that Induct is also capable of producing default rules with exceptions (Gaines, 1994, 1995) that are often more natural in their correspondence to psychological models, and runs Entail again with "EDAG" selected in Figure 7. The system generates the screen of Figure 8 which has the default rule to reject all students unless they have a high gpa, good references and a willing supervisor—a good initial characterization of the selection process.

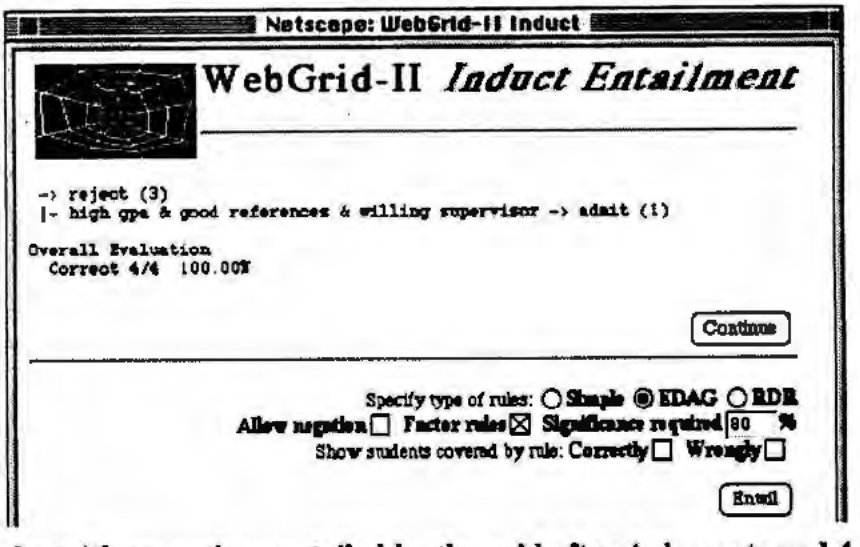

Figure 8 Rules with exceptions entailed by the grid after 4 elements and 4 constructs

3.3 Inference through an Embedded Expert System Shell

The user decides to test the entailed rules on further cases and returns to the status screen of Figure 5 which now shows 4 constructs. She clicks on the "Test" button under the list of elements and the system returns the screen shown in Figure 9 which allows her to specify a test case and click on "Infer" to see what decision is anticipated.

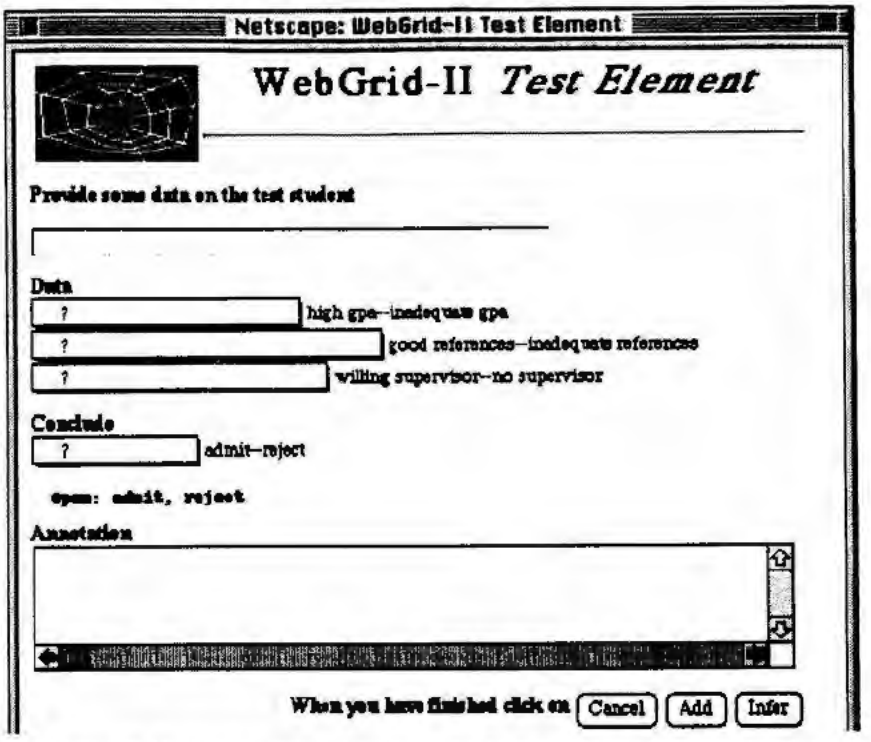

Figure 9 Entering a test case into the entailed expert system

The user does this for several similar cases and the conclusion returned is correct but when she does so for the student "lemp" as shown in Figure 10 the conclusion is "reject" when she knows the student has been accepted.

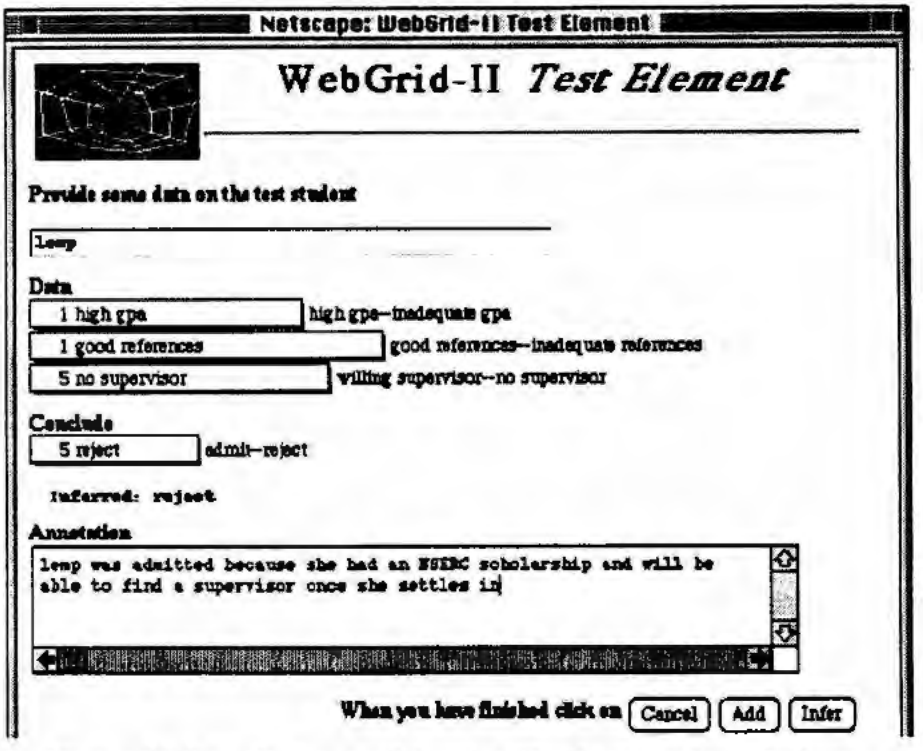

Figure 10 Conclusion anticipating the decision for a test case

The user realizes that student "lemp" was admitted because she had a scholarship, corrects the conclusion to "admit", and clicks on "Add" to add this student as a new element in the grid. The system returns to the status screen of Figure 5 with "lemp" added and the user clicks on "Add" under the construct list to enter the construct "has scholarship—no scholarship." She then clicks on "Entail" and the system generates the entailed rules with exceptions as shown in Figure 11 which now account for "lemp" also. The default rule is to reject all students unless they have a high gpa and good references when they should be accepted unless they have no supervisor and no scholarship—a more refined characterization of the selection process.

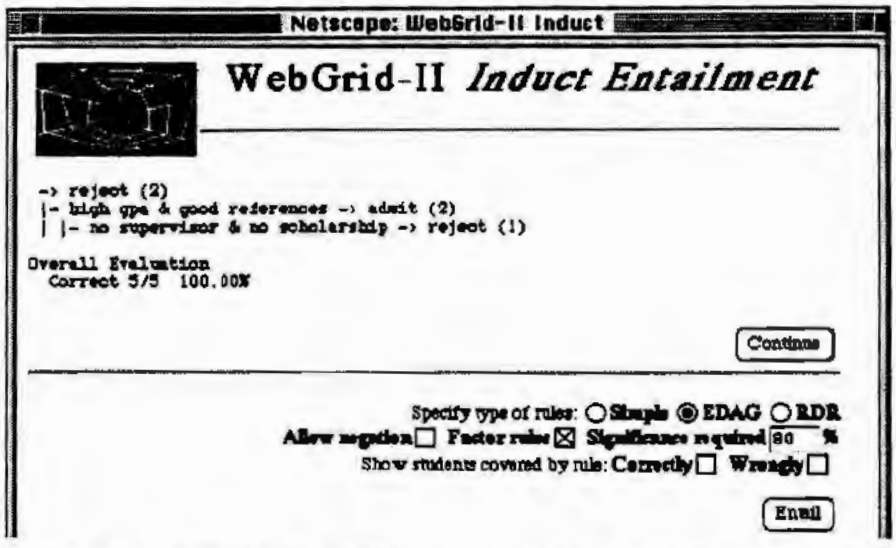

Figure 11 Rules with exceptions entailed by the grid after 5 elements and 5 constructs

The user returns to test case entry again and this time come across a student who was admitted to a new program, an industry-based software engineering degree with the requirement that the student have substantial industry background. She enters some relevant cases and constructs leading to the entailed decision rule shown in Figure 12, that students should be rejected by default unless they have a high gpa and good references and in addition have either a willing supervisor, a scholarship or are a software engineering application with industry experience.

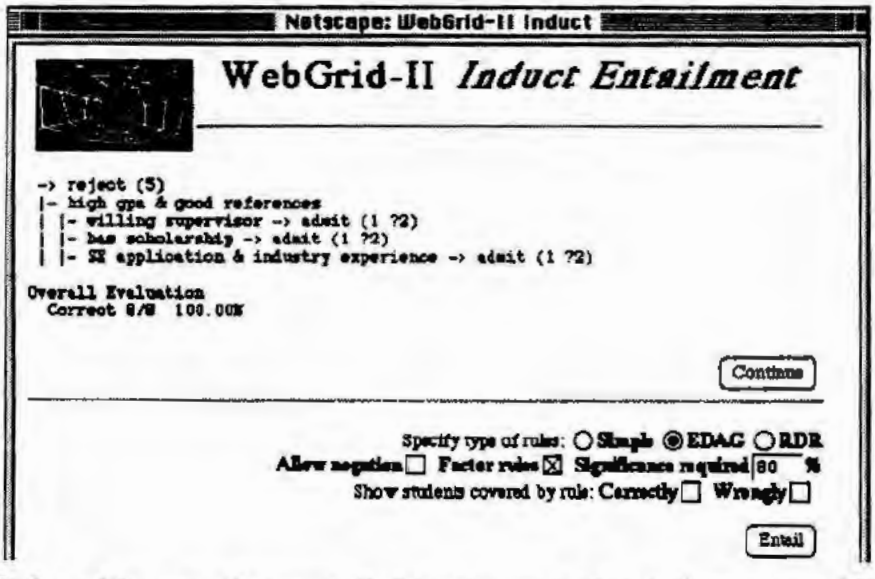

Figure 12 Rules with exceptions entailed by the grid after 8 elements and 7 constructs

Note how the decision rule entailed by the grid is becoming more refined as it encompasses more cases and also how the Induct algorithm generating it changes its structure to be as comprehensible as possible as it grows in complexity. This attempt to generate an anticipatory structure which is natural for people is an important feature of EDAG generation (Gaines, 1995). It is also noteworthy that a fairly small grid of only 8 elements and 7 constructs can be used to generate such a complex decision rule. This is because the repertory grid elicitation technique with PEGASUS-style (Shaw, 1980) feedback leads to the elicitation of typical elements and relevant constructs, allowing knowledge to be elicited highly efficiently compared with rule induction from a database where thousands of examples might be required to derive the same knowledge (Gaines, 1989).

3.4 Graphical Analysis through FOCUS and PrinCom

The grid elicited is a normal repertory grid and the normal tools for analysis and display of the data are also available. Figure 13 shows the output returned when the "Cluster" button in Figure 5 is used to develop a hierarchical knowledge model by sorting the elements and constructs so that similar ones are close together using the FOCUS algorithm (Shaw, 1980). The results of analysis are usually presented as graphic output. This is generated at the server, converted to GIF format and returned to the client where it can be examined and saved if required.

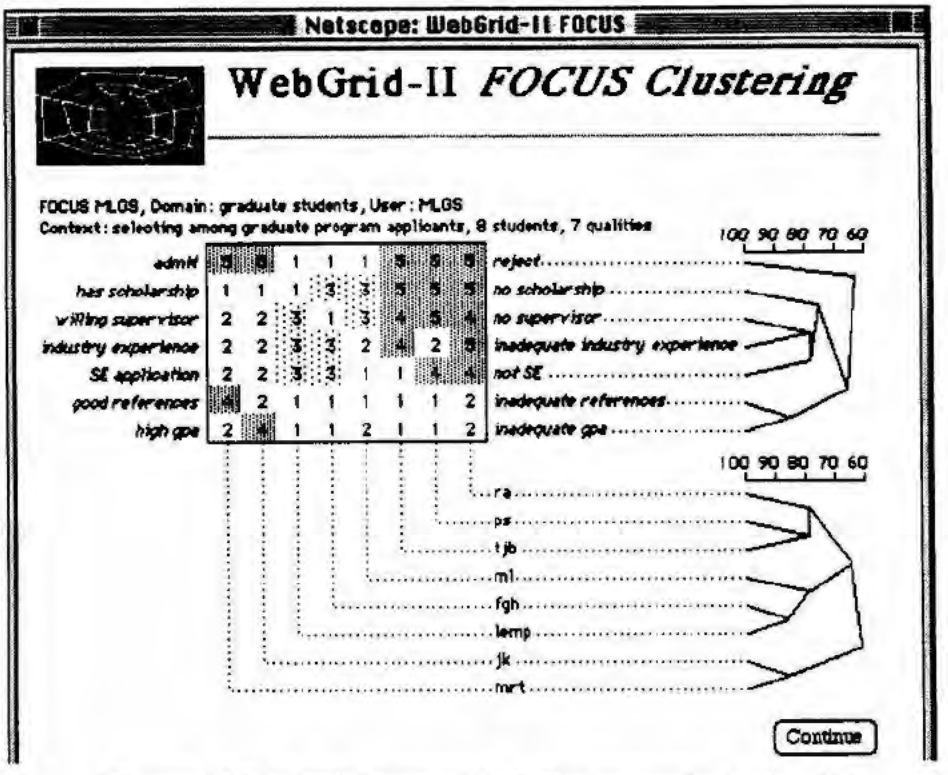

Figure 13 FOCUS hierarchical cluster analysis of grid

Figure 14 shows the output returned when the "Map" button is used to provide a principal components analysis of the grid by rotating it in vector space to give maximum separation of elements in two dimensions (Slater, 1976; Gaines and Shaw, 1980). The results of analysis are again graphed, converted to GIF format and returned to the client where they can be examined and saved if required.

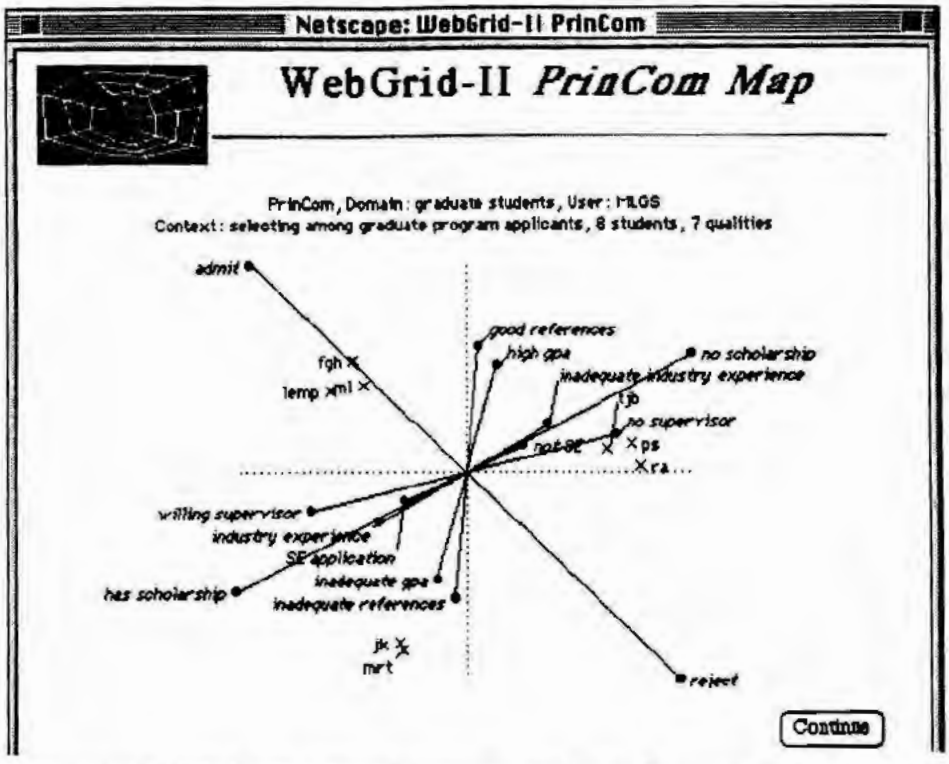

Figure 14 PrinCom map of first two principal components of grid

The knowledge acquisition server is designed to operate on a stateless basis in which data is not normally stored at the server unless it is specifically uploaded. This is achieved through the server storing all the grid data in hidden fields in the HTML document which do not display but are returned to the server along with the entered data. The user can store the data at the client machine by saving the source of the document in a file, and can continue the elicitation and analysis at any time by loading this file. The embedded data is readily extracted for use in other applications, and WebGrid-II also provides facilities to export the data in other interchange formats for use in offline programs such as RepGrid and StatView.

4 New Data Types in Grids

One of the extensions to repertory grids that has proved important in knowledge acquisition is to support the representation of constructs that goes beyond rating scales and allows data types such as categories and numbers to be represented. It is possible to do this within the framework of personal construct psychology by using Kelly's (1969) notion of a "geometry of psychological space" in which the constructs are dimensions (Shaw and Gaines, 1992). Numeric scales along a dimension are natural constructs, and categories correspond to planes slicing off part of a dimension. In practice, this PCP analysis has turned out to be important in guiding the appropriate use of different construct types in knowledge acquisition—the use of categories in arbitrary ways that does not conform to the notion of bipolar constructs generally leads to difficulties in knowledge representation.

As an example of the use of other data types, consider the grid on graduate student admissions developed in the previous section, and a test case where the student does not have a high gpa but has good references and industrial experience and will be admitted because the gpa is adequate, rather than good. The user might decide to categorize gpa as "inadequate", "adequate" and "high" to allow the representation of this situation. She can do this by clicking on the "Edit"

button at the bottom of the status screen in Figure 5 and selecting the full range of options, "Ratings + Names + Categories + Numbers" in an options editing screen similar to that of Figure 2. The full range of options will now appear whenever a new construct is being edited.

Figure 15 shows the gpa categories being added as a new construct. The screen now makes provision not only for rating scales but also for a set of categories, ordered or unordered, and for numerical rating scales of the type integer, float or date with prescribed ranges. The user has entered "inadequate", "adequate" and "high" as possible categories for gpa and has entered "gpa" as the name of the construct. The only difference the name entry makes is to shorten the process since WebGrid will generate the names "gpa = inadequate", etc, rather than have the user enter "gpa is inadequate", etc, as category names. The elements are rated on a categorical construct through popup menus, exactly as they were for rating scales.

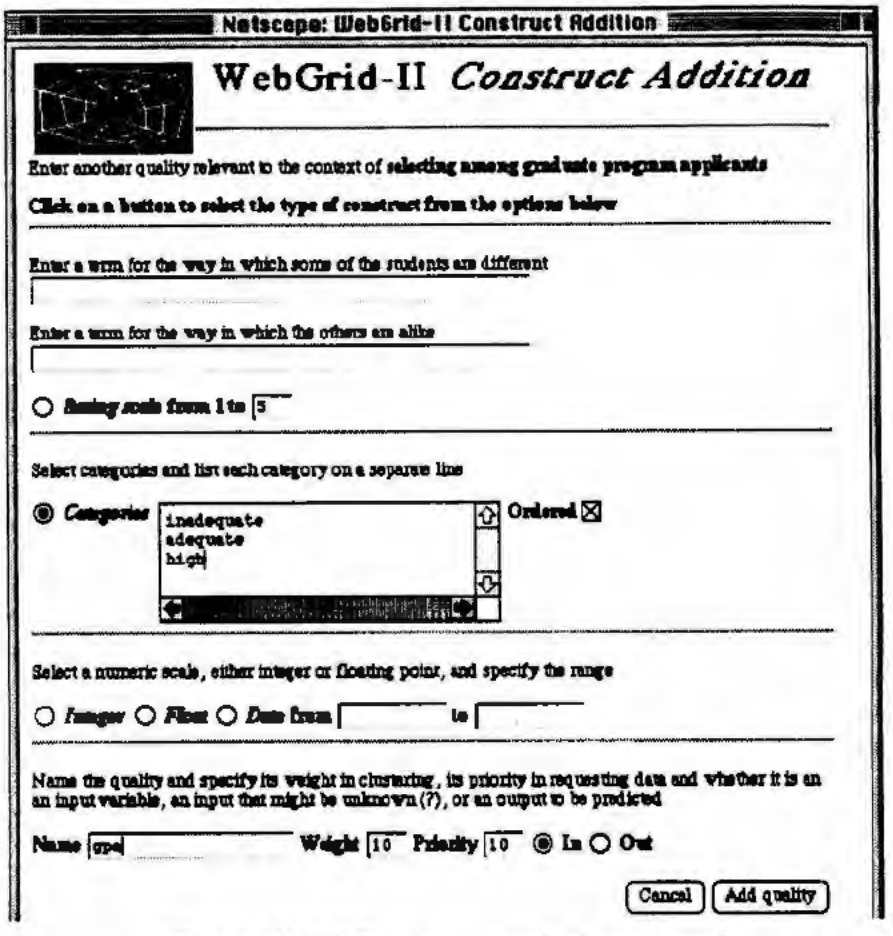

Figure 15 Adding a categorical construct

Figure 16 shows the model developed when "Entail" is run with the additional case and categorical construct. It says that students should be rejected unless they have good references and either their gpa is high and they have a willing supervisor or a scholarship, or they have made an SE application and have industry experience, except that these latter should be rejected if their gpa is inadequate. Thus, the model has incorporated the data from the additional case as an exception to an exception which is a natural pattern of reasoning for people.

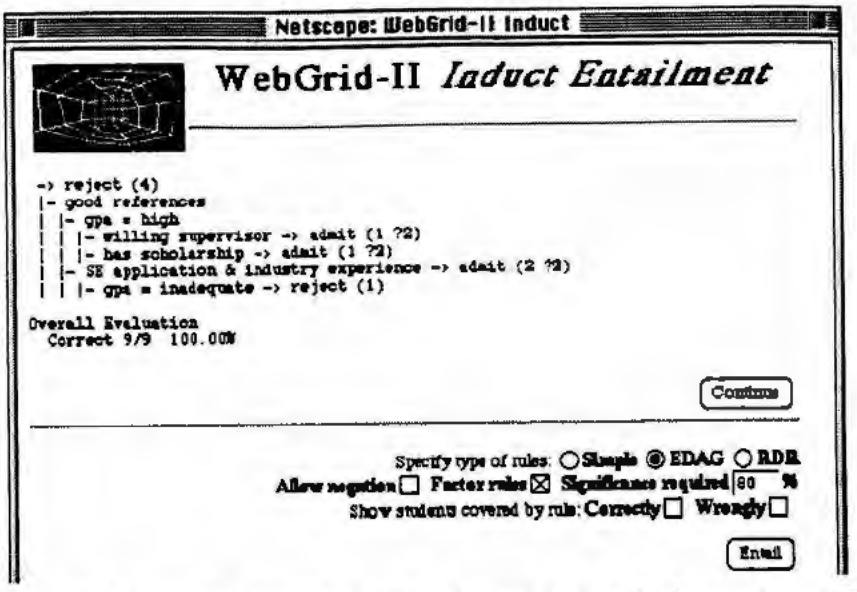

Figure 16 Rules with exceptions entailed by the grid after 9 elements and 7 constructs

In this example it would also be natural to use a numeric rating scale for the gpa which is typically on a 0 through 4 scale with up to two places of decimals. This can be entered as a floating point numeric rating scale with the categories specified as shown in Figure 17.

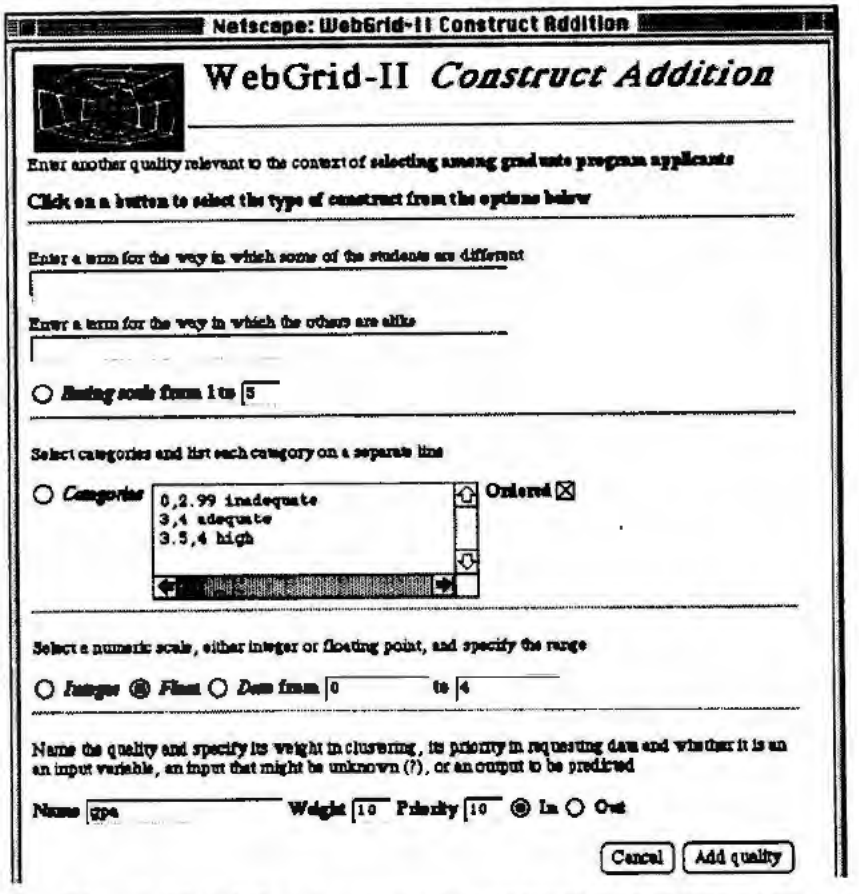

Figure 17 Adding a numeric construct with categories

The user now enters numeric values for the gpa directly in text entry boxes rather than through popup menus and the system automatically generates the classification as defined. Note that the classifications are not mutually exclusive, a gpa that is "high" is also "adequate". This is also true for ordered categories where the Entail analysis automatically groups inner values with the values above them or the values below them. This preserves bipolarity and generally leads to more meaningful models.

The extension of a repertory grid with categorical and numeric data types enables it to encompass the technical features of a computer 'database' but does so in such a way that the underlying principles of personal construct psychology are preserved. This relationship between human knowledge representation and technical data representation is a significant direction for further research.

5 Multimedia Annotation

A major feature of the World Wide Web is that it provides for the transport and display of multimedia documents containing typographic text, pictures, videos, sounds and hyperlinks to other such material. WebGrid-II provides full access to such facilities by allowing the header and trailer of each page to be customized through HTML text controlling multimedia access and presentation, and each element to be annotated in HTML. The annotation is displayed when the element is the topic of an elicitation dialog allowing, for example, images to be attached to elements. Figure 18 shows the elicitation of a construct from a triad of elements each of which has been annotated with HTML that specifies that an image should be displayed.

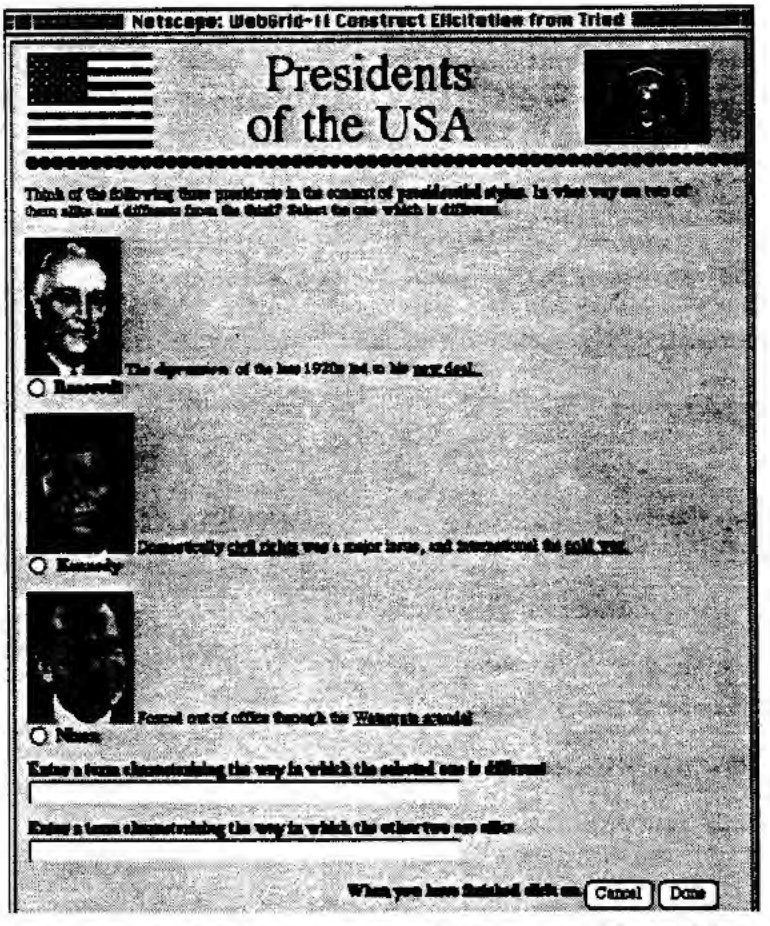

Figure 18 Construct elicitation from a triad of elements with multimedia annotation

All of the multimedia material shown in Figure 18 was entered directly as annotation by the user and did not require any change to WebGrid code. The heading with flags was entered through a text box at the bottom of the initial screen which allows a heading and a trailer to be specified that will appear on each WebGrid page. The pictures of the presidents and the links to other material on the web were entered as element annotations. Thus, WebGrid supports the hypermedia capabilities of the World Wide Web as a fully integrated capability.

6 Group Comparison of Repertory Grid Conceptual Structures

Once knowledge acquisition tools are available on the web it becomes possible to support distributed knowledge acquisition in which experts and knowledge engineers at different sites collaborate in developing a knowledge-based system. The repertory grid system described above has been extended to support the comparison of knowledge structures using an extension of our consensus/conflict/correspondence/contrast methodology (Shaw and Gaines, 1989; Shaw, 1994a). When a grid has been developed by an expert at one site, another expert at another site can develop a grid based on it, either attempting to rate the elements on the other expert's constructs to evaluate consensus and conflict, or developing her own constructs to evaluate correspondence and contrast.

Figure 19 shows the initial screen for a user eliciting a grid based on the elements in another user's grid on concepts of advanced information systems (Shaw, 1994h). When the expert has entered their name and clicked on "Done" the server generates screens for triadic elicitation and rating similar to those of Figures 3 and 4, and elicitation proceeds as before.

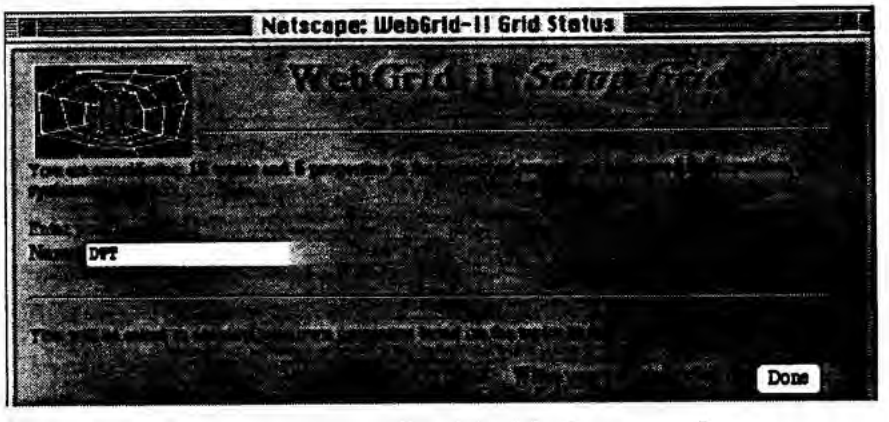

Figure 19 A user starting an "exchange" grid with the same elements as another user

Figure 20 shows the status screen after six constructs have been elicited, and it will be noted that a new "Compare" button is available underneath the analysis buttons in the lower part of the screen.

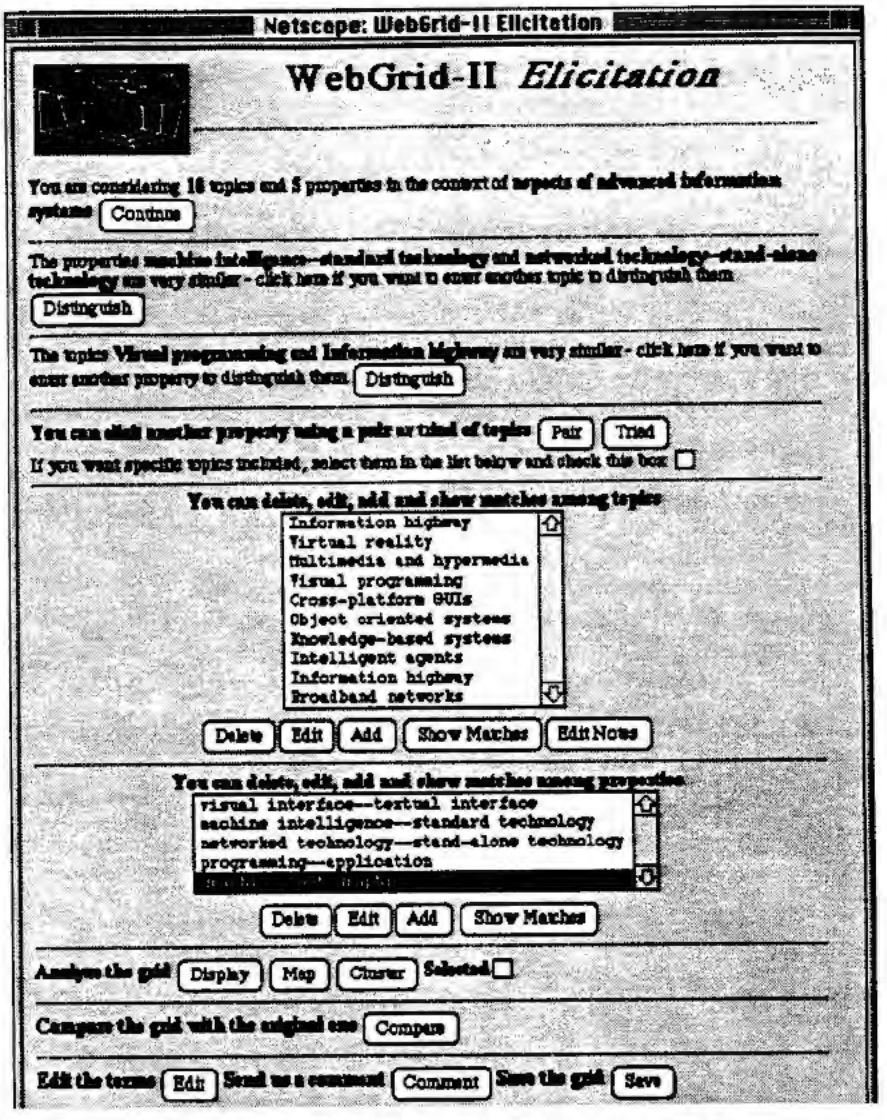

Figure 20 Status screen with compare option

Clicking on this generates the analysis shown in Figure 21 where each construct in the new grid has been matched with the closest corresponding construct in the original grid and the degree of match has been graphed on the right. This enables the users to see where they have corresponding constructs either providing different terms for the same distinction or indicating some underlying relation, for example a causal link. They can also see the constructs and elements where they do not have correspondence and about which they may have difficulty in communicating.

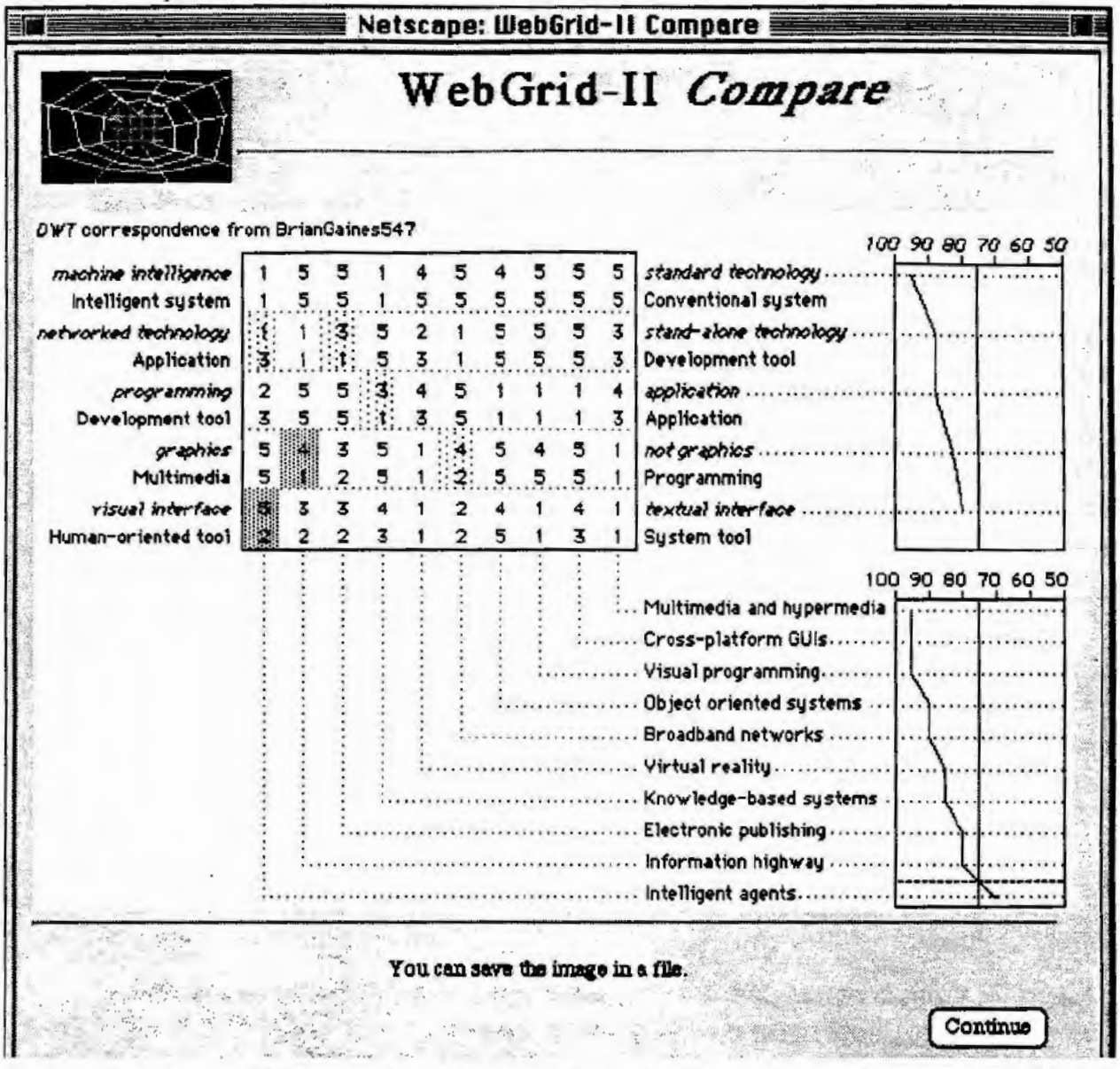

Figure 21 Comparison of grids with same elements from two users

The "exchange" methodology may also be used in which a user supplied with a grid from another user and asked to fill in the ratings. The resultant grids may then be compared using a difference methodology (Shaw, 1980) and the results presented graphically as shown in Figure 22. The users may see what contructs they are able to use in a similar way and what elements they construe in a similar way.

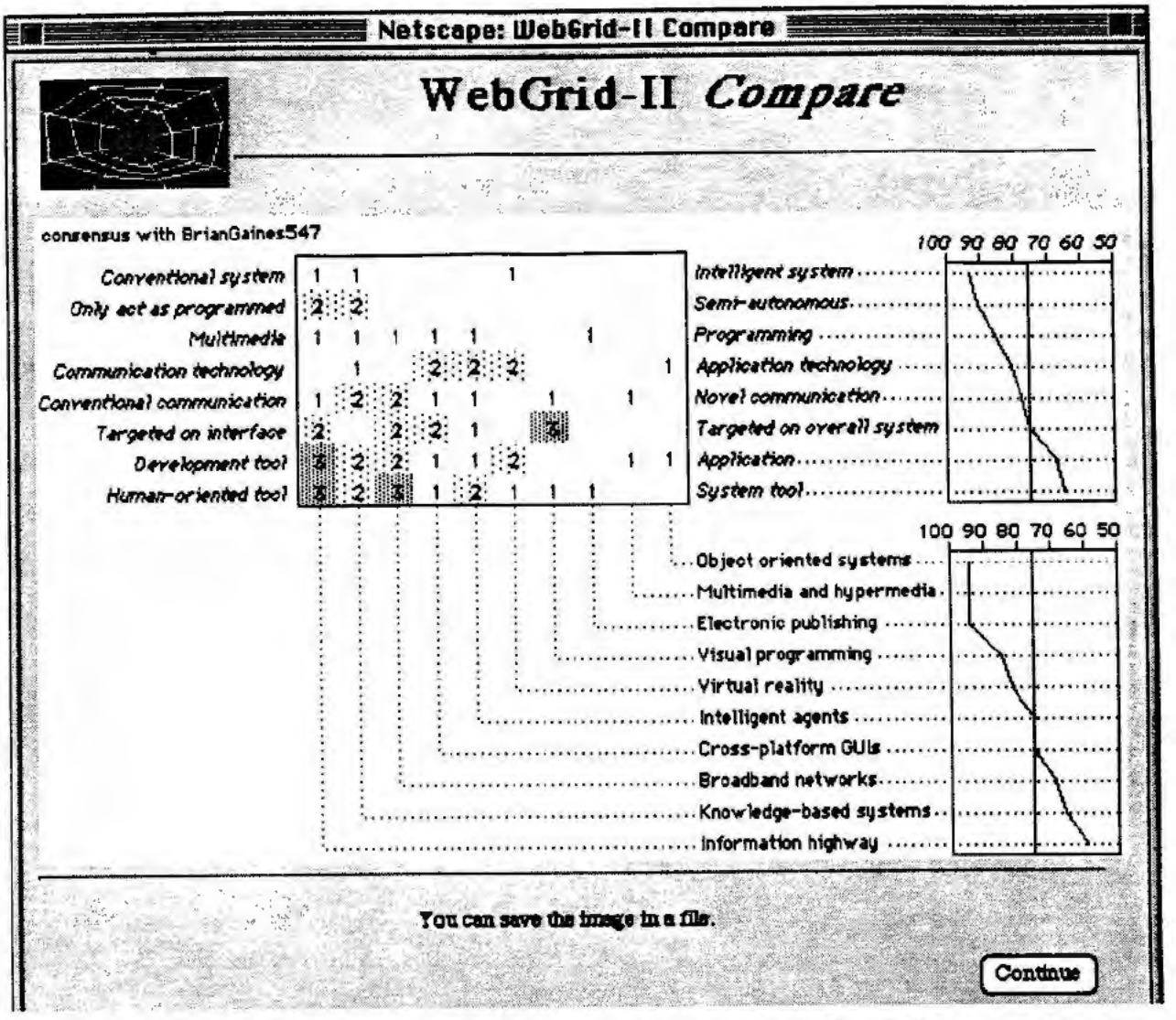

Figure 22 Comparison of grids same elements and constructs exchanged between two users

The results can be discussed through email, Internet audio and videophone, or through more conventional means. In some cases the discussion may lead to a consensus on terminology and knowledge structures, in other cases to systems combining different perspectives and in others to systems that offer advice based on different and potentially conflicting sources of expertise (Boose and Bradshaw, 1987).

7 Conclusions

The wide range of methodologies and tools developed in personal construct psychology have the potential to support knowledge-based system development across many disciplines, but access to them is currently not widely available. The technologies, tools and protocols developed for the World Wide Web make it possible to provide knowledge acquisition, representation and inference systems internationally to anyone with access to the Internet. Such knowledge-based technologies integrate naturally with other applications and hypermedia systems to provide a distributed knowledge medium capable of supporting the knowledge processes of professional communities.

This paper has illustrated what is possible through examples of the development of knowledge modeling tools operating through the web to provide knowledge acquisition, representation and inference through repertory grids. These systems were developed primarily for locally managed projects, and their availability has only been advertised through specialist list servers for the knowledge acquisition and personal construct psychology communities. However, in the period June 1995 to June 1997 WebGrid was accessed from over 5,000 different sites in some 60 countries, delivering over 200Mbytes of grid data in some 70,000 transactions. The web is essentially an anonymous medium and we do not track the activities of outside users. However, occasionally users contact us to discuss their applications and, if the local network goes down, we receive mail from serious users reporting the problem and requesting notification of when the knowledge acquisition programs will be available again. An interesting example of serendipitous use was a masters student in Holland studying operators' models of nuclear reactors who came across the grid elicitation system on the web in September 1996 used it with his experts to elicit their conceptual models, and included these in an additional chapter in his thesis for examination in October 1996—the web certainly accelerates research processes.

In conclusion, we note that the results presented in this paper demonstrate that the World Wide Web client server model operating through the Internet can effectively support the porting of existing knowledge acquisition, modeling and representation tools to operate in a distributed, open architecture environment. This opens up new possibilities for collaborative research in which the methodologies and tools developed by one group are made available on the network for use by other research groups world-wide. It enables past studies of the integration of hypermedia, knowledge acquisition and inference tools within a local environment (Gaines and Linster, 1990; Gaines, Rappaport and Shaw, 1992) to be generalized to integration on a distributed basis across wide area networks. A major advantage of this approach is that tools can be operated at the site of the originators responsible for them, and their maintenance, upgrading and monitoring can be done very much more effectively than when the software is distributed to other sites.

We envision a collaborative research program internationally in which the first objective is to make tools available across the Internet in interoperable forms, and the second objective is to develop some integrated knowledge management systems that use these tools in a variety of configurations to provide a new generation of knowledge-based systems. The deep psychological foundations of the tools in personal construct psychology provides a principled basis for developing and extending knowledge-based systems technology, and much of what is being developed for purposes of artificial intelligence research may have more general applications in, for example, clinical and educational studies.

Acknowledgments

Financial assistance for this work has been made available by the Natural Sciences and Engineering Research Council of Canada.

URLs

WebGrid can be accessed at **http://tiger.cpsc.ucalgary.ca/WebGrid/**

Related papers on WebGrid and World Wide Web can be accessed through **http://ksi.cpsc.ucalgary.ca/articles/**

References

Andreessen, M. (1993). NCSA Mosaic Technical Summary. NCSA, University of Illinois. ftp://ncsa.uiuc.edu/Web/Mosaic/Papers/mosaic.ps.Z.

Berners-Lee, T. (1993). World-Wide Web Talk at Online Publishing 1993. CERN, Geneva. http://info.cern.ch/hypertext/WWW/Talks/OnlinePublishing93/Overview.html.

Berners-Lee, T. and Cailliau, R. (1990). WorldWideWeb: Proposal for a Hypertext Project. CERN, Geneva. http://info.cern.ch/hypertext/WWW/Proposal.html.

Boose, J.H. (1984). Personal construct theory and the transfer of human expertise. **Proceedings AAAI-84**. pp.27-33. California, American Association for Artificial Intelligence.

Boose, J.H. and Bradshaw, J.M. (1987). Expertise transfer and complex problems: using AQUINAS as a knowledge acquisition workbench for knowledge-based systems. **International Journal of Man-Machine Studies 26** 3-28.

Boose, J. H., Bradshaw, J. M., Koszarek, J.L. and Shema, D. B. (1993). Knowledge acquisition techniques for group decisions support. **Knowledge Acquisition 5**(4) 405-447.

CPCS (1993). **RepGrid 2 Manual**. Centre for Person Computer Studies, 3019 Underhill Drive NW, Calgary, AB, Canada T2N 4E4.

Ford, K.M., Bradshaw, J.M., Adams-Webber, J.R. and Agnew, N.M. (1993). Knowledge acquisition as a constructive modeling activity. **International Journal of Intelligent Systems 8**(1) 9-32.

Gaines, B.R. (1989). An ounce of knowledge is worth a ton of data: quantitative studies of the trade-off between expertise and data based on statistically well-founded empirical induction. **Proceedings of the Sixth International Workshop on Machine Learning**. pp.156-159. San Mateo, California, Morgan Kaufmann.

Gaines, B.R. (1991). Refining induction into knowledge. **Proceedings of the AAAI Workshop on Knowledge Discovery in Databases.** pp.1-10. Menlo Park, California, AAAI.

Gaines, B.R. (1994). Exception DAGs as knowledge structures. **AAAI-94 Workshop: KDD'94 Knowledge Discovery in Databases**. Menlo Park, California, AAAI.

Gaines, B.R. (1995). Transforming rules and trees into comprehensible knowledge structures. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R., Ed. **Knowledge Discovery in Databases II**. pp.205-226. Cambridge, Massachusetts, AAAI/MIT Press.

Gaines, B.R. and Linster, M. (1990). Integrating a knowledge acquisition tool, an expert system shell and a hypermedia system. **International Journal of Expert Systems Research and Applications 3**(2) 105-129.

Gaines, B.R., Rappaport, A. and Shaw, M.L.G. (1992). Combining paradigms in knowledge engineering. **Data and Knowledge Engineering 9** 1-18.

Gaines, B.R. and Shaw, M.L.G. (1980). New directions in the analysis and interactive elicitation of personal construct systems. **International Journal Man-Machine Studies 13** 81-116.

Gaines, B.R. and Shaw, M.L.G. (1993a). Basing knowledge acquisition tools in personal construct psychology. **Knowledge Engineering Review 8**(1) 49-85.

Gaines, B.R. and Shaw, M.L.G. (1993b). Eliciting knowledge and transferring it effectively to a knowledge-based systems. **IEEE Transactions on Knowledge and Data Engineering 5**(1) 4-14.

Kelly, G.A. (1955). **The Psychology of Personal Constructs**. New York, Norton.

Kelly, G.A. (1969). A mathematical approach to psychology. Maher, B., Ed. **Clinical Psychology and Personality: The Selected Papers of George Kelly**. pp.94-113. New York, Wiley.

Shaw, M.L.G. (1980). **On Becoming A Personal Scientist: Interactive Computer Elicitation of Personal Models Of The World**. London, Academic Press.

Shaw, M.L.G. (1994a). A methodology for sharing personal construct systems. **International Journal of Constructivist Psychology 7**(1) 35-51.

Shaw, M.L.G. (1994b). Women, scholarship and information technology: A post-modern perspective. **Transactions of the Royal Society of Canada 5** 113-131.

Shaw, M.L.G. and Gaines, B.R. (1983). A computer aid to knowledge engineering. **Proceedings of British Computer Society Conference on Expert Systems**. pp.263-271. Cambridge, British Computer Society.

Shaw, M.L.G. and Gaines, B.R. (1989). A methodology for recognizing conflict, correspondence, consensus and contrast in a knowledge acquisition system. **Knowledge Acquisition 1**(4) 341-363.

Shaw, M.L.G. and Gaines, B.R. (1992). Kelly's 'Geometry of psychological space' and its significance for psychological modeling. **New Psychologist** 23-31.

Shaw, M.L.G. and Gaines, B.R. (1993). Group knowledge elicitation over networks. Bramer, M.A. and Macintosh, A.L., Ed. **Research and Development in Expert Systems X. Proceedings of British Computer Society Expert Systems Conference**. pp.43-62. Cambridge, UK, Cambridge University Press.

Shaw, M.L.G. and Gaines, B.R. (1995). Comparing constructions through the web. Schnase, J.L. and Cunnius, E.L., Ed. **Proceedings of CSCL95: Computer Support for Collaborative Learning**. pp.300-307. Mahwah, New Jersey, Lawrence Erlbaum.

Shaw, M.L.G., Gaines, B.R. and Linster, M. (1994). Supporting the knowledge engineering life cycle. Bramer, M.A. and Macintosh, A.L., Ed. **Research and Development in Expert Systems XI**. pp.73-86. Oxford, SGES Publications.

Slater, P., Ed. (1976). **Dimensions of Intrapersonal Space: Volume 1**. London, John Wiley.