CONTEXT-SENSITIVE VISUAL RESOURCE BROWSER

Oleksiy Khriyenko

Industrial Ontologies Group, Agora Center, University of Jyväskylä
P.O. Box 35(Agora), FIN-40014 Jyväskylä, Finland

ABSTRACT

Now, when human becomes very dynamic and proactive resource of a large integration environment with a huge amount of different heterogeneous data, it is quite necessary to provide a technology and tools for easy and handy human information access and manipulation. We believe that for the most effective information communication to the user, the content and visual representation of the content needs to be adapted based on the task the user is performing, the user goal, the context of the situation. Context-awareness and intelligence of user interface brings a new feature that gives a possibility for user to get not just raw data, but required information based on a specified context. In this paper we consider the requirements for such intelligent interface and present the first prototype of SmartInterface (UBIWARE project1) - smart adapter for human in Human-Resource communication process and intelligent through-resource browser.

KEYWORDS

Smart interface, intelligent resource visualization, context-dependent multidimensional visualization, visual search and browsing, human-computer interaction.

1. INTRODUCTION

Now, when human becomes very dynamic and proactive resource of a large integration environment with a huge amount of different heterogeneous data, it is quite necessary to provide a technology and tools for easy and handy human information access and manipulation. In the Semantic Web context, a number of specifications to express content semantics, including RDF, OWL, Topic Maps, etc have been defined. Regarding the question how to visualize this information, the crucial point is to transform abstract information structures into a visual form enabling the viewer to observe, navigate, understand and interact with the information. A solution to this problem is the application of information visualization techniques to the Semantic Web. Since the Semantic Web architecture is a complex and multi-layered building, its visualization may vary according to the type and nature of information to be graphically represented (Geroimenko and Chen, 2004). Different strategies have to be applied to visualize topic maps, ontologies, RDF schemas, RDF resources, XML or SVG documents. Resources and their connections are graphically represented as a star connected graph with labels indicating the existing relations (Catenazzi and Sommaruga, 2005). But still, we believe that for the most effective information communication to the user, the content and visual representation of the content needs to be adapted based on the task the user is performing, the user goal, the context of the situation.

Semantically enhanced context-dependent multidimensional resource visualization (Khriyenko, 2007a)(Khriyenko, 2007b) provides an opportunity to create intelligent visual interface that presents relevant information in more suitable and personalized for user form. Context-awareness and intelligence of such interface brings a new feature that gives a possibility for user to get not just raw data, but required information based on a specified context. Now it has become evident that we cannot separate visual aspects of both data representation and graphical interface from interaction mechanisms that help a user to browse and query a data set through its visual representation. Following GUN-Resource centric approach (Kaykova et al, 2005), let us consider user interfaces for context-based resource access and contextually related information retrieving. The challenging task is to create a visual interface that provides integrated

information from variety of information providers in context-dependent way. Following new technological trends, it is time to start a new stage in user visual interface development – a stage of semantic-based context-dependent multidimensional resource visualization. 4i (FOR EYE) technology is a step to achieve this goal (Khriyenko, 2007a). 4i is an ensemble of GUN Resource Platform Intelligent GUI Shell (smart middleware for context dependent use and combination of a variety of different MetaProviders depending on user needs) and MetaProviders - visualization modules/platforms that provide context-dependent filtered representation of resource data and integration on two levels (information/data integration of the resources to be visualized and integration of resource representation views with a handy resource browsing).

Context-awareness and intelligence of such interface brings a new feature that gives a possibility for user to get not just raw data, but required integrated information based on specified context. GUI-Shell allows user dynamic switching between MetaProviders for more suitable information representation depending on a context or resource nature. From other side, MetaProvider plays fore main roles: Context-aware resource visualization module that presents information regarding to specified context in more suitable and personalized for user form; interface for integrated information visualization with intelligent context-aware filtering mechanism to present only relevant information, avoiding a glut of unnecessary information; visual Resource Platform that allows resource registration, search, access and modifying of needed information/data in a space of registered resources; mediator that facilitates resource to resource (R2R) communication. Such switching and filtering process is a kind of visual semantic browsing based on semantic description of the context and resource properties. Now, when unlimited interoperability and collaboration demand data and information sharing, we need more open semantic-based applications that are able to interoperate and collaborate with each other. Ability of the system to perform semantically enhanced resource search/browsing based on resource semantic description brings a valuable benefit for today Web and for the Web of the future with unlimited amount of resources. Proposed resource visualization approach can find a place and can be utilized in various visual systems and especially in next-generation human-centric open environments for resource collaboration with enhanced semantic and context-based visual resource browsing. It can be considered as a new valuable extension of text-based Semantic MediaWiki to Context-based Visual Semantic MediaWiki. This is a good basis for the different business, production, maintenance, healthcare, social process models creation and multimedia content management.

The paper contains two main sections. Section 2 is focused on requirements for the 4I technology components. The second section 3 describes implementation of the first SmartInterface prototype for context-aware through-resource browsing and relevant information retrieving.

2. REQUIREMENTS FOR THE 4I TECHNOLOGY COMPONENTS

2.1 Intelligent GUI Shell

2.1.1 Context-based selectiveness and filtering

One of the main features of the GUI Shell is an ability to represent the information regarding to chosen context of resource visualization. Interface allows user to simply choose a context for data representation, for complex views and filtering purposes. It is not necessary to define all of the contexts manually, the system should be supplied with a module for intelligent construction of complex context from the initial one based on domain ontology, knowledge bases and history of through-resource browsing process. Concerning an intelligent filtering of data to be presented, system should select and visualize just relevant to the context resource properties. For this purpose additionally generated context is needed. And of course, to build such intelligent filtering mechanism, we need to have knowledge, ontology of resource relations and different processes descriptions. Thus, the GUI-Shell should provide handy interface for resource browsing and context generation. And if we are talking about an interface that provides dynamic switching between different views (MetaProviders), then Shell should provide a possibility to generate context from the visualization module (MetaProvider) interface. It brings a need to organize a flexible mechanism for interoperation between GUI-Shell and MetaProviders.
2.1.2 Personalization

GUI-Shell should provide user personalization features. System should remember user’s preferences regarding to different visualization contexts, views, more popular MetaProviders and etc. There are can be many MetaProviders, which are belonged to the same class of MetaProviders, but the visualization methods of certain properties may be different. At the same time system should support alliance approach, when resources are bound in some business, production or another process. In this case system should propose the MetaProviders that have registered the necessary partners.

2.2 MetaProvider

2.2.1 Integrated data representation

As a portal for registered resources, MetaProvider should support a simple resource registration process (depending on a specific of presentation view), store resource semantic profiles for further resource discovery and inter-resource communication. Such communication implies an observance of the same ontology by resources. Following the common ontology, resources have a possibility to create a correct request to each other, and get and present appropriate data. And as was mentioned before, integrated data representation means representation of the data from different resources in certain context related to a subject resource. It is a representation of filtered relevant to the certain case data.

2.2.2 Resource property visualization

Information Visualization aims to provide compact graphical presentations and user interfaces for interactively manipulating large numbers of items. Much of the work in this field focuses on creating innovative graphical displays for complicated datasets. Information Visualization allows designers to present a large amount of information using abstract representations. Geographic and scientific visualization applications usually use representations determined by the nature of the data being displayed. On the other hand, Information Visualization allows designers to choose among a palette of possible representations that fill space in a variety of ways, such as hierarchies, time lines, networks, tabular displays and the like, to produce information abundant displays. Choosing the appropriate representation(s) is challenging and research is needed to evaluate and compare different approaches (Marcos et al., 2005). Visualization techniques include selective hiding of data, layering data, taking advantage of 3-dimensional space, using scaling techniques to provide more space for more important information (e.g. fisheye views), and taking advantage of psychological principles of layout, such as proximity, alignment, and shared visual properties. Advanced interfaces also need to address the longer term process of analysis that may require annotation, history keeping, collaboration with peers, and the dissemination of results and procedures used. Faster rendering algorithms, sophisticated aggregations techniques to deal with large datasets, and novel labelling techniques are also needed, and along with careful studies of users and their needs will lead to successful visualization applications.

2.2.3 Dynamic interoperability

Interoperability among the components of the environment should be provided via metadata and common ontology. Additionally, if we consider the open environment, members of which are agent-driven (i.e. proactive, autonomous, goal-driven, intelligent and etc.) to enable communications, coordination and negotiations between each other, then MetaProvider should support inter-agent communication via semantic API.

3. SMARTINTERFACE

Work in one of the workpackages of UBIWARE project is aimed on elaboration and development of the SmartInterface – smart adapter for human in Human-Resource communication and intelligent through-resource browser. As mentioned above, SmartInterface is presented by GUI-Shell (central browser of the system) and remote distributed visualization modules - MetaProviders.
3.1 SmartInterface interaction model

Following figure (see Figure 1) presents a full interaction model of the system. On the first step, user searches for initial resource to start through-resource browsing process. Based on keywords that describe type and content of a resource, GUI-Shell searches resources via accessible databases or with a help of MetaProviders that have own data storages. Then, user can choose proper resource from a list of found resources. On the second step, GUI-Shell browses ontology and gets a list of relevant resource visualization contexts. Based on specified context for current resource, GUI-Shell searches for appropriate visualization modules (MetaProviders). There are can be number of them (with different implementations, sets of features, etc.) When certain MetaProvider has been chosen, it sets up communication with GUI-Shell and, based on specified resource and context, decides what the relevant resources (in this context) are and what data needed for proper visualization is. On the fourth step, MetaProvider starts to retrieve necessary information from own databases, through communication with GUI-Shell or other MetaProviders. When all (at least necessary part) the data is collected, MetaProvider performs visualization in Resource Visualization Area of the GUI-Shell. Fifth interaction step between GUI-Shell and MetaProviders is reserved for feedback when MetaProvider informs the GUI-Shell about selected resource. Thus, whole through-resource browsing cycle ends and starts here.

![SmartInterface interaction model](image)

For the next year we are going to implement whole system fully based on the first year version of UBIWARE Platform. Communication between GUI-Shell and MetaProvider is planned to be organized based on Agent communication. MetaProviders themselves will decide (reason) what is the necessary data and will perform retrieving of it with a help of the GUI-Shell or via own data retrieving channels. But, in current version, to avoid non-reusable implementation of communication between GUI-Shell and visualization modules, we put a description of input data (necessary resources’ properties) for MetaProviders to their descriptions. Thus, GUI-Shell collects required data and sends it via the request on MetaProvider loading.

3.2 GUI-Shell

GUI-Shell is presented by Html-page and remote server part that plays role of search engine and performs all necessary complex calculations. Main Html-page contains five areas:

- **Resource Search Area**: Based on keywords, which describe a resource type (class of resources) and a content of a resource, user gets a list of corresponding resources;
- **Resource Description Area**: Here, properties of current (currently selected) resource are displayed to the user. In current version of the prototype, user may just observe resource properties and cannot make any changes to them;
- **Visualization Context Area**: Here system provides a list of visualization contexts for currently visualized resources;
- **MetaProviders Area**: Depending on chosen resource visualization context, a list of appropriate visualization modules (MetaProviders) is presented for user in this area;
- **Resource Visualization Area**: This is an area, where visualization page of chosen MetaProvider is loading and performs.
  
  Another important part of the GUI-Shell is server part of it. As a search engine, it performs several tasks:
  - based on keyword of resource type and resource content descriptions, returns a list of matched resource back to Html-page;
  - parses Ontology and returns a list of correspondent visualization contexts relevant for a class of selected resource;
  - returns a list of appropriate MetaProviders based on specified resource (resource class) and visualization context.

  To organize dynamic search on-the-fly, we applied AJAX technology\(^2\) during the GUI-Shell development. Thus, this technology allows us to build dynamic SmartInterface and increase the interactivity, speed, functionality and usability of it.

### 3.3 MetaProvider – “Member-Of visualizer”

On current stage of the SmartInterface concept development we are not focused on the development of various visualization modules. On the first step we decided to develop the most frequently used visualization of the general “member-Of” relation. This was enough to perform a case of browsing through several resources in different contexts with the same visualization module. The case includes a visualization of an organization (some research group) in “personnel, stuff…” and “related projects” contexts, visualization of chosen project in the context of “project consortium” (or “project management board”) and further browsing of chosen organization from the set of project consortium members. Even this simple case shows us ability of the system to help user to find, for example, a person, which is somehow related to the certain research group as a management board member of some project leaded by this group, if no other information is available.

Concerning development of a simple MetaProvider (“Member-Of visualizer”) that visualizes resource in contexts were “member-of” property plays main role, we decided to utilize X3D\(^3\) related technologies with the purpose to visualize it in more natural for human way. X3D is a royalty-free open standard file format and run-time architecture to represent and communicate 3D scenes and objects using XML. To provide communication with X3D scene, we utilized SAI (Scene Access Interface) that allows a programmer to change or build X3D worlds, and AJAX technology to allow on-the-fly scene modification. In current implementation the only strict requirement for MetaProvider development is “callFunction” input parameter. This parameter contains the name of call-back function of the GUI-Shell which MetaProvider should call and send the ID of current resource as a parameter.

“Member-Of visualizer” is a simple MetaProvider that visualizes a resource and related to it resources via “member-of” upper-property. For example, members of an organization (group) or projects leaded by this organization, project consortium or management board members, etc. MetaProvider is developed based on X3D technology to present resource in more demonstrative for human form. As the one of the best representation forms, the image based representation has been used to visualize a resource. Person is visualized via his/her photo, organization and project – via logo. That is why, required data for such visualizer is: resource ID, image (photo, logo, etc.), image width and height of subject resource; and same data set of other related resources. “Member-Of visualizer” is a JSP page that gets request from GUI-Shell and, based on incoming parameters, utilizes server part of the MetaProvider to dynamically create an appropriate X3D file to be loaded in imbedded X3D plug-in. Visualizer applies the same visualization approach for all the resources. This is a 3D scene with an image (photo or logo of subject resource) in the center of it. All the other related resources (their images) are located around of the main one form a circle/disk. Interface allows rotation of the circle/disk to get the best view point. Pointing on an image causes highlighting of it and makes correspondent resource selected. The click on any, related to the subject one, resource results its visualization. To be able communicate with GUI-Shell, MetaProvider gets a callback function in a request from it. Each time, when user selects a resource on the MetaProvider, this function is called and GUI-Shell gets a correspondent resource ID to show the resource properties and continue resource visualization process.

---

\(^2\) AJAX technology - [http://www.w3schools.com/AJAX/default.asp](http://www.w3schools.com/AJAX/default.asp)

\(^3\) X3D technology - [http://www.web3d.org/](http://www.web3d.org/)
3.4 Useful features

With a purpose to make interaction with a SmartInterface more user-friendly and do not demand a lot of manipulation from user, we defined default visualization contexts for resource classes and MetaProviders for certain visualization context. User is able to specify favourite visualization module for certain resource in certain visualization context. Also, we implemented a smart technique for automatic selection of a visualization context. The logic is be based on a history of visualization contexts and resources that user has browsed/visualized previously. This context ranking technique allows us to sort a list of visualization contexts in more appropriate order for user and gives him/her a hint for next logical step. Thus, it becomes a smart search system that leads the user in proper direction/way.

4. CONCLUSION

With a purpose to prove the concepts under 4I (FOR EYE) technology, we elaborated and developed functional prototype. We highlighted the most important requirements for such an intelligent human resource browser and implemented initial main parts of it. As a goal for correspondent workpackage of the second year UBIWARE project, we are going to elaborate probably the most important part of 4I vision, which can be called “context provision”. Especially when considering a human, presenting information on a resource of interest alone is not sufficient - information on some “neighbouring” objects should be included as well, which form the context of the resource. What is important is that in different decision-making situations, different contexts are relevant: depending on the situation the relevant neighbourhood function may be e.g. physical spatial, data-flow connectivity, what-affects-what, similar-type, etc. The ability to determine what type of context in right one for the situation and collecting the information that forms the context of that type for a specific resource is central in 4I vision. Proposed resource visualization approach can find a place and can be utilized in various visual systems and especially in next-generation human-centric open environments for resource collaboration with enhanced semantic and context-based visual resource browsing. It can be considered as a new valuable extension of text-based Semantic MediaWiki to Context-based Visual Semantic MediaWiki. This is a good basis for the different business, production, maintenance, healthcare, social process models creation and multimedia content management.

ACKNOWLEDGEMENT

This research has been performed as part of my PhD studies and UBIWARE project in Agora Center (University of Jyvaskyla, Finland) funded by TEKES and industrial consortium of Metso and Metso Automation, ABB, Fingrid, TeliaSonera, Inno-W and Hansa Ecuras. Also, I am very grateful to the members of “Industrial Ontologies Group” for fruitful cooperation.

REFERENCES


