TECHNISCHE UNIVERSITÄT WIEN
VIENNA UNIVERSITY OF TECHNOLOGY

DISSERTATION

The Concept of *Connecting Ontology* and its Exploitation for Knowledge and Information Presentation for People with Special Needs

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Doktors der technischen Wissenschaften unter der Leitung von

O.Univ.-Prof. Dipl.-Ing. Dr.techn. A Min Tjoa
E188/1 Information and Software Engineering Group
Institute of Software Technology and Interactive Systems
Technische Universität Wien

und

a.Univ.-Prof. Dr. Klaus Miesenberger
Institut Integriert Studieren
Johannes Kepler Universität Linz

eingereicht an der
Technische Universität Wien
Fakultät für Informatik

von

Muhammad Shuaib Karim
Matrikelnummer: 0327350
Obermüllnerstrasse 2C / 6 / 408
A-1020, Wien, Österreich

Wien, September 2007

Dedicated to my dear father Shams, M.D. (Late) and my dear mother who encouraged me to do whatever I wanted to pursue in my career, and to my dear wife and lovely children Arfa, Maryam and Umar who waited for so long for the completion of my thesis and always cheeringly supported and prayed for me.

Zusammenfassung

Die rasante Entwicklung und die wachsende Verwendung von *Informations- und Kommunikationstechniken* in den letzten Jahrzenten hat zahlreiche Möglichkeiten zu Verbesserung der Lebensqualität für Benutzer des *World Wide Web* hervorgebracht. Die Fortschritte in der Datenspeicherung ermöglichen es, die gesamten Arbeit eines Benutzers am Computer, die während eines Lebens getätigt werden, zu speichern. Dies führt zu einer riesigen Menge an Informationseinheiten in verschiedenen Formaten von den unterschiedlichsten Quellen. Dies beinhaltet vielen Herausforderungen, die im Gebiet von Wissensmanagement und Wissenspräsentation für Endbenutzer behandelt werden. Die Daten sind nicht verwendbar, wenn die Verbindungen zwischen den einzelne Information und den Gedanken und Ereignissen im Leben des Benutzers fehlen. Die richtig verbundenen Informationen mit Hilfe von Metadaten stellen ein digitales Tagebuch des Benutzers dar. In den Wörtern von *Vannevar Bush*, ist es wie *eine intime ergänzung zu seinem Gedächtnis*.

Zweifelslos würde dies den Benutzern in vielerlei Hinsicht nützlich sein, falls die Darstellung den Fähigkeiten und Präferenzen des Benutzers entspricht. Die sich über die Zeit ändernden Fähigkeiten und Präferenzen eines Benutzers gestalten diese Aufgabe schwierig. Die Aufgabe wird erschwert, falls die Informationen für Benutzer mit speziellen Anforderungen, mit physischer oder geistiger Beeinträchtigung, präsentiert werden müssen. Auf Grund des Fehlens eines generischen Ansatzes ist das *Accessibility Konzept für Computersysteme*, die mit wenig Aufwand für die speziellen Anforderungen konfiguriert werden können, weder sehr populär, noch sehr weit verbreitet. Das Hauptaugenmerk liegt jeweils auf dem normalen Benutzer. Auch die gegenseitigen Vorteile für Benutzer mit speziellen Anforderungen und jene ohne wurden nicht erkannt. Dies erschwert die Entwicklung von Produkten für Benutzer mit Beeinträchtigungen und blockiert andererseits die Möglichkeiten von Vorteilen, die auch für Benutzer ohne Behinderungen nützlich sind. Daher gibt es die Notwendigkeit die Strategien für eine Gewinnsituation beider Seiten anzupassen.

Man erreicht dies, indem die kontextuellen Bestandteile des Systems passend modelliert und verbunden werden. Dabei wird die Beeinträchtigung des Benutzers als eine der relevanten Komponenten betrachtet. Die vor kurzem aufkommende Technologie des semantischen *Webs* stellt Werkzeuge und Techniken zum Modellierung und Verbinden von heterogenen Informationsressourcen im *Web* zur Verfügung. Das Konzept von *"Connecting Ontology"* wurde hier eingeführt um heterogenen Domänen basierend auf den *Regeln des semantischen Webs* zu verbinden. Die Regeln repräsentieren das implizite Wissen über die Domänen, und die daraus gebildete *"Connecting Ontology"* ist direkt nutzbar um Vorschläge unter Berücksichtigung der speziellen Anforderungen des Benutzers zu geben. Es ist außerdem möglich, wie bei jeder anderen Ontologie, weitere abstraktere Szenarien zu erfüllen. Ein Testfall zur Verbindung von Daten über die Beeinträchtigung von Benutzern und Eigenschaften über Benutzeroberfläche wird präsentiert um die Validität und Wirksamkeit des Ansatzes zu zeigen. Zu diesem Zweck wird das vorgestellte *Accessibility Framework* eingebunden, das mit dem *Visualisierung-Subsystem* des Prototypens zusammen wirkt. Ebenfalls können andere kontextuelle Komponenten des Systems miteinander für ein *Usable* and *Accessible* system verbunden werden.

Der Hauptbeitrag dieser Doktorarbeit bildeten das Konzept von *"Connecting Ontology"* für die Verbindung von heterogenen Domänen und die Ontologie für Daten über die Beeinträchtigung von Benutzern und Eigenschaften über Benutzeroberfläche. Die zukünftige Richtung beinhaltet die Entwicklung von *"Connecting Ontology"* als *semantisches Web Services*, die welche automatische Verbreitung der Ontologie über Daten über die Beeinträchtigung von Benutzern und Eigenschaften über Benutzeroberfläche unterstützt. Eine weiter Entwicklung wäre die semantische Repräsentation verschieder Visualisierungstechniken, sodass die Visualisierung nicht im Voraus gewählt werden muss und der Benutzer online auswählen nach den geläufigen Informationssemantik während der Navigation.

Abstract

The exponential growth in the development and the subsequent usage of the *Information and Communications Technology* over the last decade has provided numerous opportunities to the human users of the *World Wide Web* for improving their quality of life. The advancements in the data storage technology have practically made it possible to store the user's computer activities carried out during his / her entire lifespan. This has given rise to massive amounts of information items in multiple formats which may originate from heterogeneous resources. Nonetheless, this is also intertwined with many issues to cope with in the domain of knowledge management and its presentation to the end users. The lifetime information is not very usable if the associations of information items with each other and with user's thoughts and life events are not established. The properly connected user's lifetime information items using the meta data are like the user's digital diary. In the words of *Vannevar Bush*, it is like *an enlarged intimate supplement to his memory.*

Certainly, this would be very helpful for the user from many dimensions when presented according to his capabilities and preferences. Keeping in view the changing nature of capabilities and preferences over time, the task is a challenging one. The situation is aggravated when the information is to be presented for the users with special needs or impairments, who are physically or mentally challenged in carrying out their activities. Due to lack of a generic approach, the concept of providing *accessibility* for people with special needs is neither very popular nor widely practiced in the industry for providing *accessible computer systems* which can be customized for several types of impairments within reasonable resource constraints. Thus the major concentration is always towards the normal users, the dominant segment of the population. Also, the mutual benefits for the users with or without any impairments are not very well recognized which not only hampers the production of accessible products for people with disabilities, but also blocks many avenues which could even be more useful for users without any disability. Thus there is a need to adapt the strategies with a Win/Win situation for both.

This can significantly be achieved if the contextual components of the system are appropriately modeled and interconnected, while also considering the *User's Impairments or Disabilities* as one of those components. The recently emerging *Semantic Web Technology* provides us the tools and techniques to model and interconnect the heterogeneous information resources over the *Web*. The concept of *Connecting Ontology* is introduced to connect the heterogeneous domains based upon the *Semantic Web Rules*. The rules represent the tacit knowledge about the domain, and the *Connecting Ontology* thus formed is directly usable for giving suggestions according to *user's impairments*. It is also exploitable like any other ontology for fulfilling more abstract scenarios. A test case to connect the *User's Impairments Data* and the *User Interface Characteristics* is presented to show the validity and efficacy of our approach. For this purpose, the proposed *Accessibility Framework* is incorporated which works in coordination with the *Visualization sub-system* of our prototype. On similar lines, other contextual components of the system can be connected with each other for a *Usable* and *Accessible system*.

The major contributions of this thesis are the concept of *Connecting Ontology* for connecting heterogeneous domains, and the ontologies for the *User's Impairments Data* and the *User Interface Characteristics* which are beneficial in their own right. The future directions include the deployment of the *Connecting Ontology* as a *Semantic Web Service*, automatic population of the *User's Impairments* and *User Interface* ontologies, and the semantic representation of different visualization techniques so that the visualizations are not chosen in advance, but could be available for on-line selection by the user depending upon the prevalent information semantics during navigation and browsing.

Acknowledgement

I am grateful to God the Almighty for giving me this wonderful life and utmost courage to follow the path of knowledge exploration and to achieve the milestone of my Doctoral Thesis.

I am most thankful to my adviser, Professor Dr. A Min Tjoa for providing a conducive working environment for both independent and collaborative research. Although four years is a pretty long duration, I wish I could have spent more time in his group of self motivated researchers and excellently managed office environment. All of them were very helpful during my research. I owe special thanks to Andreas Rauber, Alexander Schatten, Monika Lanzenberger, Martina Seidl, Maria Schweikert, Sonja Willinger, Natascha Surnic, Gia Phu Liu and Michael Schadler.

I am also grateful to my second adviser Dr. Klaus Miesenberger for his guidance and encouragement. The diverse nature of the thesis required an out of the box thinking for its solution originating from the combination of distinct areas of *Accessibility for People with Special Needs* and *Knowledge Management*. I feel myself lucky to work closer to the advisers from both of these areas. That helped me a lot for a successful combination of both of these domains for steering my way through to completion.

I have to say thanks to my brother Zubair Kareem, M.D. Phy-Neurologist for the long intermittent discussions in order to develop a taxonomy of a representative set of user's impairments and its potential use for rehabilitation purposes, which is most probably the next phase of my research.

My heartiest gratitudes are for my friends in the *SemanticLIFE* project and in the office with whom I worked for so many years and learnt a lot, both academically and otherwise. We had some very fruitful discussions about the ontologies and the lifetime store which helped a lot in refining my thesis. I am especially thankful to Khalid Latif, Shah Khusro, Ahmed Mansoor, Hoang Huu Hanh, Khabib Mustofa, Amin Andjomshoaa, Ferial Shayeganfar, Georg Pölzlbauer, Carl Rauch and Horst Kargl. My special thanks goes to my colleague Stephan Strodl for translating the thesis abstract in German. During the course of my studies, I had very good academic discussions with numerous people at different forums. That also helped me a lot in my research. My sincere thanks to all of them. I am also thankful to my friend Shaheen Ashraf who was my room mate during all these years and provided such a nice company.

It was not easy to start this big endeavor at the beginning. I am obliged to all my teachers and friends who gave me a lot of support and stimulation for going ahead with my Doctorate.

PhD is a long and lonely path to traverse which needs optimism and steadfastness through out the entire duration. I am thankful to my parents and all my family members who were extremely supportive during my studies while I was away from home and my family. I could not have succeeded without support from all of them. Especially, I have to say a big "THANK YOU" to my wife, my daughters Arfa and Maryam, and my son Umar for their patience and courage to face everything excellently during my absence. It was a tremendous moral support for me throughout these long years.

For providing a marvellous opportunity to study in Austria, the efforts of the Higher Education Commission of Pakistan, the Austrian Exchange Service (ÖAD) and the ASEAN-EU University Network Initiative are commendable. I am especially thankful to the HEC Pakistan for the generous funding for my Doctoral studies, and to my parent institute Quaid-i-Azam University, Islamabad for sparing me to pursue my studies abroad.

Vienna, September 2007
Muhammad Shuaib Karim

Contents

List of Figures

List of Tables

Listings

Chapter 1

Motivation and Overview

Over the *World Wide Web* the amount of published material and *Web* based communications supporting a diverse range of content types and interaction devices, has been increasing at an incredible pace during the recent years. This is a huge distributed knowledge base open for Internet users. Still, the foreseeable exploitation of this information is largely undermined due to inability of human users to link heterogeneous information resources, and explore and retain the intricate associations that exist between different aspects of information. The goal is unsurmountable if it is left to the human users alone to read, interpret, associate and deduce the information over the *Web*, thus stagnating the process of acquisition and management of knowledge. Thanks to the *Semantic Web*, which provides the ways for formal information representation of *Web* resources, thereby enabling the *Software Agents* to automatically interpret and process the published information for presentation to the end users according to their needs, capabilities and preferences (Berners-Lee, 1999). This is also the main focus of ***Web Accessibility****, which means that people with disabilities can perceive, understand, navigate, and interact with the Web, and that they can contribute to the Web. The user's disabilities or "Impairments" may be cognitive, physical, or multiple which further complicates the matter* (WAI, 1997). The increased difficulties for users with special needs are noticeable at two levels. Firstly, the *User Interaction* with the system is compromised due to their physical disabilities and secondly, the *Impairments* being closely related with *Human Cognition*, may influence the process of *Knowledge Management* itself.

These issues are not only concerned with the *Web* based software. The popularity and pervasiveness of the *Web* has also embraced the legacy information systems which typically used to function in isolation from the *Web*, such as *Enterprise Information Systems*, *Medical Diagnostic Systems*, and *Personal Information Management Systems*, just to name a few. Now these systems could equally benefit from the rich potential of *Semantic Web Technology*. The investigation of semantic relationship between the functional entities in an *Information Management System*, and their formal representation is of high value for automation purposes. In such systems the information flow revolves around the key entities such as the various functional domains, information items generated from these domains, associated resources and their roles, and a number of system imposed constraints. The functional entities must interact with each other for providing meaningful results to user queries (Holzinger et al., 2007). The results are transformed into appropriate *User Interfaces (UI)* according to *the device and the user profiles*. In order to make the system *Universally Accessible*, the special needs of the users, must be taken into account (WAI, 1997).

The *User Interaction* with the system presumes the *User Profile* to be an integral component which corresponds to *User's Impairments* and *Capabilities*. Both of these fall under the umbrella of *"Digital Human Modeling"* which is by now, well recognized in the domains of engineering, robotics and medicine as is evident from many projects and initiatives such as (Tollis and Ayache, 2007) (Grail, 2002) (Carruth et al., 2007) and (Dzaack and Urbas, 2007). However, the inspiration to explore and incorporate its effective usage in *Information Management Systems* is relatively new. This fact is also highlighted by Professor Takeo Kanade during the recent HCI International Conference 2007 in Beijing, especially the prospective benefits of modeling the *Human Capabilities* and its exploitation in *Quality of Life Technology (QoLT)* (Kanade, 2007). For improving the user's quality of life, the consensus in the community is to provide the *"right"* amount of assistance and convenience. In other words,

Assistance to be provided = User's Needs - User's Capabilities ....(1)

Since the *Impairments* should be considered as the integrative part of *Capabilities*, our approach of modeling the *User's Impairments* is in line with the above standpoint. Thus we can say alternatively as,

Assistance to be provided = User's Needs + User's Impairment's Needs - User's Capabilities of (1)

The point of caution here, is to provide just the sufficient amount of help, nothing more and nothing less. Because the out of proportion support results in a never ending loop between the stake holders' investments and gains.

The interconnection of the meaningful relationship between the semantics of the *impairments* and other components such as the interaction devices, user interfaces, and the information items, is certainly an advantage. For example, suggesting suitable font sizes according to user's visual acuity, suggesting colors according to user's particular colorblindness, suggesting options density in a menu according to user's cognitive limitations, suggesting suitable widgets according to the performance of user's motor functions would be very beneficial.

Semantic Web Technology has the right potential for the conceptual modeling of the domain of discourse at different levels of abstraction as well as its formal representation using *Description Logic (DL)*. Exploiting *Semantic Web Technology* the domains of *Impairments* and *UI* are represented formally and then rules are created to connect these. In a similar way, other entities in an *Information Management System* can also be interconnected ensuring a controlled and optimized information flow in the overall system.

1.1 Research Questions

The research questions I intend to investigate are as follows:

- Can *Semantic Web Technology* be used for providing a *Generic Accessibility* solution base for people with special needs?
- How far is it justified to assume that the investment on *UI* alone could provide maximum possible *Accessibility* for people with special needs?
- Is the sought-after approach exploitable towards diversity in general, and integration of *Information Management Systems* in particular?

1.2 Proposed Solution

Our mechanism for providing *Accessibility* is influenced by the *Data Integration* approach that calls for definition of separate schemas (*Ontologies*) for the *Contextual Components* and then connecting them together based upon user defined rules. The connection or association of two heterogeneous domains, considered as tacit knowledge, is often hard coded into the systems. As a first step, the *Connecting Ontology* approach advocates for the need and benefit of having, preferably, the separate base ontologies for different *Contextual Components* in an application. Then in second step, the base ontologies are connected at a higher level of abstraction. Consequently, the individual knowledge bases are managed independent of each other, and still being able to work in coordination due to the *Connecting Ontology* which is populated automatically from the participating ontologies and the associated rules. The rules, essentially representing the tacit knowledge, are managed interactively by the user.

In summary, an effort has been made to automate the experiences or the tacit knowledge of the user, which is a step forward from the current state of the art in *Knowledge Management* and *Accessibility* implementation efforts.

1.3 Scope and Limitations

The focus of this thesis is rather broad and generic owing to the concept of connecting the heterogeneous information domains which in principle, is applicable to almost any area of *Knowledge Management*. The importance of semantically interconnected but independent knowledge units is emphasized rather than mixing up multiple knowledge units into a few, with a lot of rich but complicated semantic relationships.

The benefits of *Impairments* domain for optimizing the *User Interaction* with the system are highlighted as a showcase example for proving our approach. Ontology definition for satisfactorily covering the corresponding domain is not a short-term but a long-term and intensive activity that requires a clear application focus and consensus of the community (Gruber, 1995). Since the objective is to show the working of *Connecting Ontology*, therefore it was not the intention to exhaustively represent the *Impairments* and *UI* domains. However, the related ontologies can be enhanced on the indicated lines using the knowledge available from medical domain like (UMLS, 1993), (OBO, 2007), and usability of user interface components such as (Northover and Wilson, 2004), (Fowler, 1998) and usability resources such as (Nokia, 2007) and (GNOME, 2006). The gathered results are post-processed programmatically for adapting the *Cascading Style Sheet (CSS)* (Bos et al., 1998) on the fly using Style Sheet parser (van Kesteren ed., 2007). Ideally, instead of connecting programmatically there would also be an ontology for *CSS* attributes which would fit into our framework for connecting with other ontologies based upon some rules.

There are different variants of *Web Ontology Language (OWL)* for modeling the ontology such as *OWL Full*, *OWL Lite* and *OWL DL* (McGuinness and van Harmelen, 2004). Each of them has some restrictions and provides varying levels of expressiveness. *OWL Full* is the most expressive one but is not yet supported by the *Reasoners*. We found *OWL DL* to be the best choice available mainly because of the availability of *DL Reasoners*. It facilitates the formal representation of semantics without compromising too much on expressiveness.

Finally, the concept of *Connecting Ontology* is clearly distinguished from apparently similar notions such as mapping, integration, alignment or fusion of ontologies (Noy and Hafner, 1997), (Kalfoglou and Schorlemmer, 2003). By virtue of resolving ambiguities within existing terminological and conceptual entities the later are concerned with knowledge reorganization, whereas the former is more about knowledge creation because it is explicitly associating the ontological entities with one another based upon the tacit knowledge.

The effects of changes in participating ontologies on the *Connecting Ontology* could become crucial and thus worth investigating too. However, this issue is not addressed in this thesis.

1.4 Structure of the Thesis

The rest of the thesis is organized as follows:

- **Background Work:** Owing to the diverse nature of the topic, a number of issues are covered in this part. These include the concept of *accessibility* and its contextual components for the *Web 1.0*, different projects related with *accessibility* over the *Semantic Web*, and data integration approaches which could be useful for providing *accessibility.* Then a brief overview of the *Semantic Web Technology*, which could be useful for our proposed approach later in the thesis, is given. At the end, we describe briefly our prototype *Personal Information Management System, SemanticLIFE* (Ahmed et al., 2004), which is an open source *Java* based system and is developed using the *Semantic Web Technology.*

- **User Interface Design Considerations:** This describes in detail the activity which was carried out for gathering data about user requirements necessary to design a usable and accessible system. A study was conducted about the end user tasks, and the adaptation of *UI* standards for developing the different components of our system by different group members while conforming to a homogeneous interface and interaction style.

- **Ontology Design Patterns for Accessibility:** These help to identify the objectives and conceptualize the scope and components of the application and the related ontologies (Gangemi, 2005). A *Generic*

Accessibility Pattern is defined followed by a number of its specializations and generalizations to cater for different user interaction scenarios for helping the users with various types of impairments.

- **Connecting Ontology:** It describes the link between two heterogeneous ontologies. More specifically, it describes the linking of heterogeneous entities (concepts, relations and properties) across two ontologies.

  The benefits of the approach are highlighted, and the issues which must be tackled to develop the *Connecting Ontologies* are described. These include the state of availability of tacit knowledge, different approaches to extract the rules behind those, and their representation. The *Connection Rules* are used to formally describe the connection between the entities in two ontologies (Karim and Tjoa, 2007). The intent is to automatically generate the *Connecting Ontology* upon execution of rules.

  Later in this part, an *Accessibility Framework* is proposed for providing accessibility at a generic level using the notion of *Connecting Ontologies*. It offers a comprehensive and generic approach for producing *Accessible* software products (Karim et al., 2007). The essential components are the base ontologies of the contextual components of the application, their interconnection with each other including the interconnection rules and associated services for their management.

- **Connecting Impairments and User Interface Ontologies:** A case study about the usage of our proposed method is presented. It shows how the *UI* is adapted according to the prevalent *User Impairments Profile* by using rules. The results are also indicative of potential benefits of the approach in other domains.

  The *Impairments* data related with user's physical and cognitive disabilities is represented as an *OWL-DL* ontology. The semantics of the *Impairments* are concentrated around their *taxonomic structure*, the Gestalt *perception cues*, and the affected *capabilities*. Meaningful property names are used to establish the relationship between them. The instances of the above three concepts are valued against a predefined measurement scale. Besides, for each *Impairment* the related body parts and their relative positions are also represented.

  Similarly, a trivial ontology of *UI* in context of *usability* of user interface components is made. It describes the containment hierarchy of the *UI* components, and the associated semantics to see if a component is only an output widget, a selection widget or a navigation widget.

The selection being "boolean" or "multi-valued" is also represented. For each *UI* component, usability measures are described in terms of representative attributes. The attributes are valued according to a predefined scale. We also specify the usage sense of *UI* components such as "functional" or "aesthetic".

The above two ontologies can be used independently for any useful purpose such as for *e-Learning* and *Medical Diagnosis*. Additionally, by executing user defined rules the *Connecting Ontology* is obtained. It describes the suggestions about appropriate *UI* components for a specific *Impairment*. The *Connecting Ontology* thus obtained is further exploitable like any other ontology.

- **Exploitation of *Connecting Ontologies* - Motivating Scenarios:** Here we present a few motivating scenarios where our approach can be usefully applied. The first scenario is about a person equipped with a high capacity photo capturing device with the possibility to take snapshots of user's activities and the context around him in real time with or without user intervention. The goal is to help the user based upon the annotations of the pictures. This can be helpful for the user in many ways during his business meetings and in his general life as well.

  The goal of the second scenario is to choose the most suitable visualization based upon the prevalent semantics of the data to be visualized and the task to be executed. The base sub-system for visualizing the user's information items is developed. It is implemented as a visualization pipeline consisting of different modules for instance retrieval from the triple store, data transformation, aggregation and presentation to the end user as a *Timeseries Visualization* synchronized with a *Browser View* with the possibility to filter and control the views.

- **Results and Discussion:** This section describes the contributions made in this thesis, and a number of in progress and anticipated future works. A critical evaluation of the goals is made at the end in order to determine how far those have been achieved.

Chapter 2

Background and Related Work

The diverse nature of this thesis renders it necessary to address the problem from more than one angles. Therefore, the areas related with the current status of *accessibility over the Web*, the *user's impairments*, the *user interfaces*, and the *Semantic Web Technology* are investigated. An overview of our prototype testbed is also presented at the end of the chapter.

2.1 Accessibility for Users with Disabilities and Accessibility for All

Accessibility for the people can be improved in many ways, such as by improving physical access (people with mobility problems require information on accessible hotel rooms, lifts, ramps), sensory access (people with hearing or visual impairments need tactile markings or hearing augmentation) and communication access (people having difficulties with written text, vision, speech, and language) (Pühretmair, 2004).

The relationship between the provision of accessible software products for users with disabilities and all other users, is often confusing and is a cause of half-hearted efforts to incorporate accessibility. Their mutual benefits are also not very well recognized in the industry which not only hampers the production of accessible products for people with disabilities, but also blocks many avenues which could even be more useful for users without any disability. The accessibility descriptions by the Standards Organizations and the related forums are worth reading for a broader understanding of the problem:

> *Content is accessible when it may be used by* ***someone with a disability*** (WAI, 1997).

> *The set of properties that allows a product, service or facility to be used by* ***people with a wide range of capabilities****, either directly or in conjunction with assistive technologies. Although 'accessibility' typically addresses users who have a disability,* ***the concept is not limited to disability issues*** (ANSI, 1997).
>
> *The usability of a product, service, environment or facility by* ***people with the widest range of capabilities*** (ISO, 2003).
>
> The (ITAW, 1998) categorically *requires Federal agencies to make their electronic and information technology accessible to* ***people with disabilities****.*
>
> Whereas, the European Union's Council Resolution (EU, 2002), and (WAI, 1997) **include the people with old age** along with people with disabilities.

From above, the three noticeable aspects about the term *Accessibility* are:

- It is not only related with people with popular disabilities
- It is not only related with user interface issues
- It is not related with specific age group

The vagueness to place the *Accessibility* on either side results in bad design decisions which influence the subsequent product developmental steps, and diminished *Return On Investment* at the end. Accessibility and usability of the information technology products and services bear good results for all the users and not only for the users with disability (Pühretmair and Miesenberger, 2005). Indeed most of the products which were originally designed for people with special needs are in active use by all, such as television remote control and voice synthesizers. Their use and popularity is mainly related with specific interaction settings or interaction constraints and the context in which they operate which closely mimic to those of a person with special needs (Obrenovic et al., 2007).

2.2 Contextual Components of Accessibility

Accessibility implementation efforts are generally focused towards typical user impairments and a handful of prevalent devices for a particular user scenario. Whereas, there are numerous factors in the prevalent context which can leverage the overall accessibility to its full potential, only if those are orchestrated together. Different types of interaction devices, varying user needs & impairments, and the specifications of the task at hand are some significant components of the context. User interfaces and visualizations are the means to carry out the user tasks in a particular context. Additionally, every visualization is not *a priori* suitable for every type of information entity which is a combination of data and the related semantics for its description. In short, there are multiple factors which can play their part towards improvement. In absence of a common and unifying approach the industry has little choice in abstaining from legacy and are therefore not very successful in producing universally accessible software (Martín González-Rodríguez and Pérez-Pérez, 2003).

In (Obrenovic et al., 2003),(Obrenovic and Starcevic, 2004) *Universal Accessibility* as a *Multi-modal Design Issue* is discussed. An effort is made to combine the concepts of *Human Computer Interaction* and *Universal Accessibility* thus narrowing the gap between them. It describes a generic *Universal Accessibility Framework* which works in coordination with the human characteristics, the interaction constraints (another sense for the term *Impairments*), and the modalities. The human characteristics are sensing, perception, motor skills, linguistic skills and cognition. The interaction constraints are related with the user, the device, social and environmental factors. The modalities are interaction styles such as textual presentation, hand movements, visual menus and speech interaction. The concept of communication channels is used which is described as a suitable combination of human characteristics, specific interaction settings, and the appropriate modalities. In our opinion, the approach of modeling the necessary components is in right direction. However, it still lacks some essential contextual components of *Knowledge Management* domain such as the tasks. In addition, the model is encoded as an ontology in *RDF* and *RDFS* (Obrenovic et al., 2007) for giving support in learning and design activities. However, the ontology modeling is not concrete enough to be used directly in software. In contrast, our approach of modeling the ontology in *OWL DL* would be helpful in multiple areas such as learning, design and also the code automation.

GADEA (Martín González-Rodríguez and Pérez-Pérez, 2003) is a *User Interface Management Expert System* for adapting the *UI* in real time based upon the user activities captured by localized *Agents*. It consists of three main modules namely *DEVA - Dialog Expert Valuator for Adaptation*, *ANTS - Automatic Navigability Testing System* and *CodeX - Code eXplorer*. *DEVA* is used to create interactive dialogs which are customize able by the user in real time. The dialog consists of meaningfully chosen interface widgets which are based upon *Usability Metrics* calculated from the characteristics about the data, the tasks and the user impairments. *ANTS* is used to update the *User Model* for the applications hosted by the framework. *CodeX* provides the communication between the server and the host application. It consists of methods which are associated, or in other words hard coded, against the widgets defined under *DEVA*. This makes it possible to dynamically choose the code, based upon changes in interaction dialog. The classical techniques of *Fuzzy Logic* are used to implement the rules. The goals to achieve are very close as ours. However, we employ ontologies using *Semantic Web Technology* between ontological models for *Impairments* and *UI Characteristics* which makes it possible to reduce the coding. For example, the usage of *Description Logic* variant of *OWL* allows by default to infer the class-subclass relationship and transitive dependencies from the model.

(Abascal-Gonzalez et al., 2003) described the *USERfit* methodology for generating the *Usability* and *Accessibility* specifications which can be helpful for the product design. The main characteristics exploited by the methodology are related with the product functions, potential users and developers, and the activities to be performed over the product. The outcome of our approach in terms of *Impairments* and *User Interface* ontologies could be helpful towards the automation of this methodology.

2.3 Accessibility Over the Web

In order to understand the possibilities available today for making the *Web Accessible*, it is useful to discuss the essential *Web* components and *Accessibility Guidelines* already in place for the first generation of the *Web*, that is the *Web 1.0*.

2.3.1 Necessary Components

As described by the *Web Accessibility Initiative* (WAI, 1997), the different components which have to interact with each other for providing an *Accessible Solution* for *Web* based applications are:

- *Information content* present in a Web page or Web application
- *User agents* like browsers, media players, programs running as part of user agents including the assistive technology products
- *Assistive technology* like screen readers, alternative keyboards, switches, scanning software
- *Authoring tools* like web content editing tools such as *HTML* and *XML* editors, tools to save and transform the documents in *Web* format, tools

for producing multimedia contents for the *Web*, tools for management of layout such as *CSS* Style Sheets, and tools for the management of *Web* sites

- *User profile* information such as user's preferences and capabilities
- *Accessibility evaluation tools* like *HTML* validators, *CSS* validators, evaluation and report language
- *Web developers* such as designers, coders, authors, which may also include people with special needs

The guidelines which bind them together are as follows (see Fig. 2.1):

- *Web Content Accessibility Guidelines*(WCAG) (Chisholm et al., 1999): Their purpose is to make the *Web Content* accessible for people with disabilities. It is in fact the core set of guidelines which also influence the other guidelines mentioned below. At present, the *Web Content* is no more the traditional static *HTML* pages, but a versatile combination of multimedia contents such as audio, video, dynamic web contents, animations, Weblogs. Each has its own content, structure and style which has to be taken care of by *WCAG*.
- *User Agents Accessibility Guidelines*(UAAG) (Jacobs et al., 2002): The user agents developed according to these guidelines, are expected to be accessible. Since, the *User Agents* are required to interface with the *Assistive Technology*, therefore, on one hand the implementation of these guidelines provide convenience for the *User Agents* such as a browser, and on the other hand embody several mechanisms which can be usefully exploited by the *Assistive Technology* such as a screen reader.
- *Authoring Tools Accessibility Guidelines*(ATAG) (Treviranus et al., 2000): These are meant for developers of the *Web Authoring Tools*. The purpose is to guide the developers for developing accessible interfaces for the authoring tools, as well as, designing and developing the tools in such a way so as to inspire and help the *Web Content Authors* to produce accessible contents using these tools.

It is also to mention that most of these guidelines cater for a few stereotypical disabilities or impairments. Moreover, due to the innate visual presentation philosophy and mostly the keyboard based interaction mechanisms for *Web 1.0*, we see a major focus on vision and motor impairments in these guidelines, and thus significant success record for the users with these impairments.

Whereas, there is a multitude of impairments one may encounter during the entire life span which pose a barrier in accessing the information. For example, even for people with vision related disabilities, mostly total blindness is assumed, whereas, there are many types of vision problems such as low vision, color blindness, problem in which the person can not recognize the images even his own. Other impairments such as persons with memory problems and mobility impairments are not significantly addressed by these guidelines (Poulson and Nicolle, 2004).

The implementation of existing guidelines, work well for people with vision related disabilities provided the information is well structured. For static textual information contents on web pages this strategy is already practiced but still there are unresolved issues. Currently the mapping of impairments with accessibility guidelines is hard coded either directly in HTML documents or in style sheets. The problem is severe where one has to navigate the ever changing information like on the web and explore the hyperlinks with possibility of each time going into a different navigational structure.

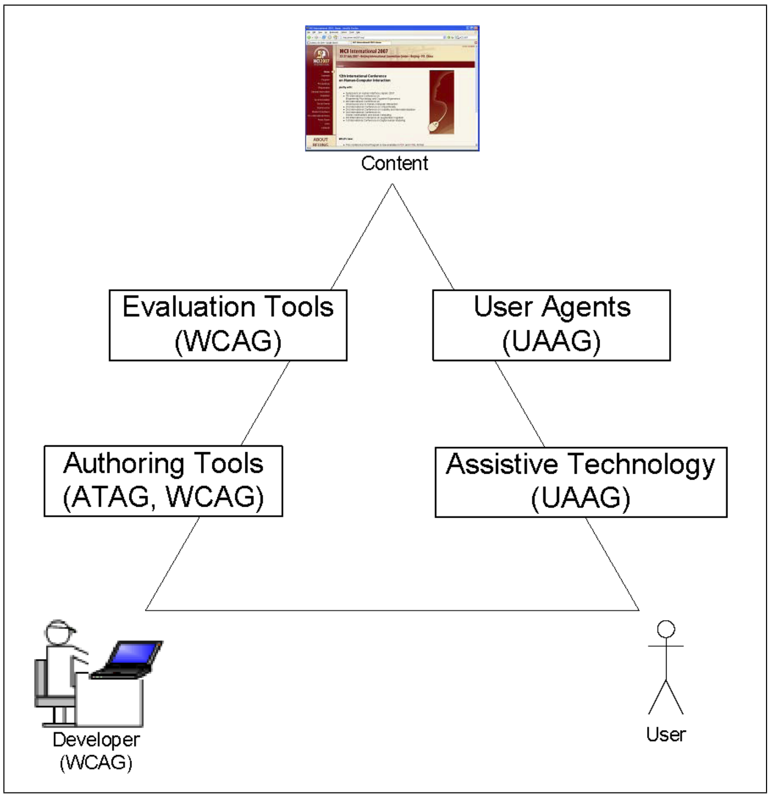

Figure 2.1: Web Accessibility Components and Associated Guidelines

2.3.2 Semantic Web Accessibility

People have used the recently emerging *Semantic Web Technology* in various ways for improving access to knowledge. The *Haystack* project (Karger et al., 2003) has successfully used *RDF* for designing a component architecture that provides rich and uniform *UI*. It consists of four parts, i.e. layout, informative, decorative, and view. The *UI* components are populated based upon the ontology contents, the meta data for whom is stored in *RDF*. The semantics of information items to be presented are associated with appropriate view parts in *RDF*. Here we can incorporate our accessibility ontology while implementing components. The possibility to click at any point on the interface for launching context sensitive queries is described in (Quan and Karger, 2004), while specifying the essential features required for a *Semantic Web* browser. This philosophy is used while designing the *UI* for the *Haystack*. It can further be enriched by the provision of task oriented semantics. This would lead to the possibility of using the most suitable visualization metaphor depending upon the changing nature of the tasks while doing searching and seeking over the heterogeneous information space like on *Web*.

Ontologies are used in (Plessers et al., 2005) for providing accessibility in context of web page annotations for conveniently navigating the visual structure of web pages. Their ontology consists of semantics about mobility (travel objects such as way points, orientation points, and travel assistants), authoring (header, logo, label, heading, footnote, section) and context (information seeking, surveying, orientation, navigation, browsing). The *WAI* provides guidelines (Chisholm et al., 1999) to encode some of the above information semantics but not all. Also, the accessibility is provided only for users with vision related impairments.

The recent work by (Harper and Bechhofer, 2005) for increasing accessibility of web pages, suggests an approach to encode the semantic information of the page directly into the page itself by introducing lightweight markup, without compromising the creative activity of authors and designers. An ontology is created representing the meaning of data in *XHTML* meta tags and then encoding this meaning into the data. This way, the relevant *CSS* remains unaffected, while the semantics become implicit part of data. It works for pages with available *CSS*. Only simple instantiations with property assertions is possible. Usage of meta tags reserved for other purposes is not a stable solution. But it highlights the possible *XHTML* enhancement with new tags for including semantics.

Different kinds of user policies (navigation, exploration, presentation, etc.) are introduced in (Encelle and Baptiste-Jessel, 2004) for making the presentation and interaction of *XML* contents accessible. The user policies are supposed to take into account the users physical and cognitive impairments by encoding the relevant information in *XML* style sheets. This is a nice and practical approach for keeping separate the original document contents and the user preferences. This can nicely be combined with Haystacks design to improve accessibility.

An organized set of steps is described in (Sheth and Avant, 2003) for designing ontological-driven semantic applications such as our prototype system *SemanticLIFE*. An additional step could be to incorporate accessibility using ontologies.

The *WWAAC* project by the European Union tried to integrate the assistive technology, the web, and the signs and symbol language used traditionally by people with communication problems for text interpretation. The accessible *UI* complexity is highlighted as a challenging issue to tackle due to conflicting needs by different users. The ontologies are used to exchange semantics between the concepts databases of symbols and the assistive technology. The framework is composed of *Concept Code Definitions* (a plain concepts list), *Base Reference Ontology* (concepts mapped from *WordNet* to the used symbols), and *Complementary Reference Ontology* (specifies missing concepts). Different user groups have been successfully tested against this framework in (Nicolle et al., 2004). The proprietary ontologies like *Assistive Technology Ontology* need to communicate with the framework by mapping against Reference, and Complementary ontologies using programming interface. The role of *User Impairments* is fixed in this framework. Our *Impairments Ontology* can be used for associating concepts from *Concept Code Definitions* to specific impairments so that presentations are customized at run time.

The human disease is conceptualized around the *Type* (disorder types), the *Symptom* (signs/indicators), the *Cause* (genetic/environmental), and the *Treatments* (surgery, drug therapy, physiotherapy, etc.) by (Hadzic and Chang, 2005). For an overview, see the Fig. 2.2. The therapy can be extended to incorporate the notion of interface therapy or adaptation for convenience and rehabilitation of the users with disabilities. The concepts of our interest are type (impairment type with associated properties to determine the severity of the impairment), and the treatment in context of *UI*. The detail is provided in Chapter 6, where *Impairment Ontology* and *User Interface Ontology* are connected to provide suggestions for *Interface Adaptation*. It is to note that our effort of *Interface Adaptation* is not meant to treat the disease of the user, as is the concept of *Disease* and *Treatment* in Fig. 2.2. Our goal is to provide convenience to the user in carrying out the information management tasks *in an analogous way*.

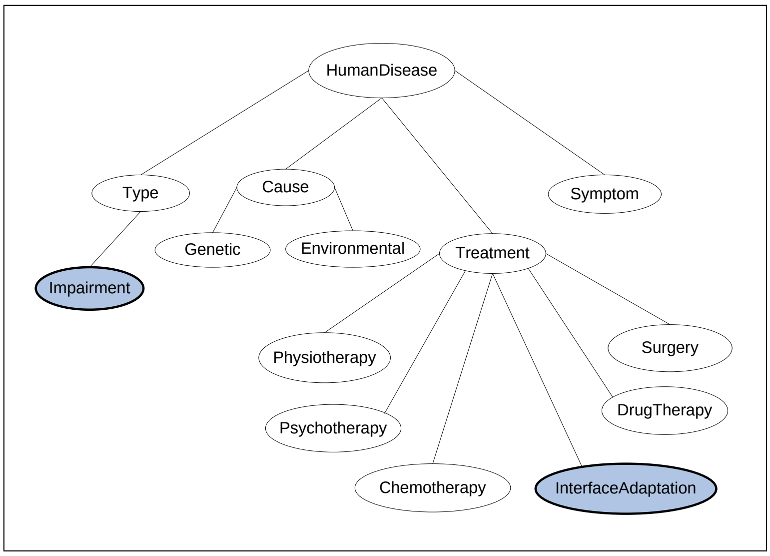

Figure 2.2: Overview of Human Disease Ontology

2.4 Data Integration Approach for Providing Accessibility

The resolution of semantic heterogeneity over the web is the key to several information integration issues [1]. The concept of *Connecting Ontologies* using the *Semantic Web Technology* is still under investigation to connect *heterogeneous domains*, especially for improving accessibility that might also be very useful for applications integration in various domains. The term has been used in different contexts though. However, due to the best of our knowledge it was never fully exploited to connect heterogeneous domains, especially for improving accessibility.

Dr. Chen and colleagues (Chen et al., 2004) highlighted the usefulness of connected knowledge in the medical field and described the initial results where ontology of a medical domain can be seamlessly scaled and integrated with ontology of another domain. The consequence was the foundation of *Semantic Web Health Care and Life Sciences Interest Group* (HCLSIG, 2004).

(Haller et al., 2006) describes a method to connect external business processes and internal work flow processes. They have used *LAV (Local-As-View)* data integration approach (Lenzerini, 2002) to map the two models. *LAV* approach provides a uniform query interface using a global mediated schema to be defined independent of the data sources. Its relationship with the data sources is then made possible by connecting the global schema with

[1] http://www.ontologymatching.org/ (5th September 2007)

specialized views for each data source. Looking at the *Accessibility Framework* (Fig. 5.2) there is a conceptual similarity with our approach where *Query Interface* is realized using a global schema and the *Connecting Ontologies* are acting as mediated interfaces between global schema and the data sources (ontologies and instances for context components). However they are connecting process workflow models pertaining to "similar domain", which is not the case in our work.

The recent work described in (Sheth et al., 2006) is an example of how *OWL* (Bechhofer et al., 2004) in combination with *RDQL* rules (Seaborne, 2004) is employed for connecting heterogeneous ontologies in an electronic medical record application. More specifically, the concepts in *SNOMED*[2] are linked with an ontology containing drug / medicine related concepts such as drug classes, their interactions, allergies and formularies. The purpose is to improve the situation of disputed medical insurance bills caused due to inconsistencies in coding schemes used during diagnosis (like *ICD9CM*[3]) and the corresponding list of approved medical procedures as permitted by insurance companies. The semantic annotations are applied in *XML* files which allow to use on one hand the available technologies like *XSL*, *XPATH* (Clark and DeRose, 1999), and rules interpretation by *RDQL*. On the other hand, since the domain is not modeled in *OWL*, therefore the sophisticated reasoning capabilities and convenient query languages like *SPARQL* can not be used. Our approach is towards modeling the domain in *OWL DL*.

The work about Contextualizing Ontologies (Bouquet et al., 2004) shows the mappings of *GALEN* medical ontology with Tambis genetic ontology by aligning both with the *Unified Medical Language System* (UMLS, 1993). Bridging rules using *C-OWL (Context-OWL)*, are defined for mapping the individual concepts or concept expressions belonging to ontologies in similar domain of discourse. In contrast, our approach of connecting ontologies provides interconnection between concepts in varying domains of discourse.

Data integration can be formally described using *Horn Clause* rules and *F-Logic* rules (Angele and Gesmann, 2006) with some benefits of expressiveness using *F-Logic*. However, the *Semantic Web Open World Reasoning* does not fit very well with *F-Logic* which is *frame-based* and influenced by object-oriented paradigm (Tetlow et al., 2006) thus involving risks of incompatibility and undecidability.

The usage of rules for forming inferred set of triples from different *OWL* models representing heterogeneous data sources has been a recent accomplishment by *Oracle*, as pointed out in (Lopez and Das, 2007). Their scala-

[2]http://www.snomed.org/snomedct/index.html (5th September 2007)
[3]http://icd9cm.chrisendres.com/ (5th September 2007)

bility and performance claims with billions of *RDF* triples are very encouraging for the adoption of *Semantic Web Technology* by the industry, especially in applications like *Personal Information Management* and *Bio Medical Domain* where the data accumulated over the years is huge. The major benefits due to the representation in form of triples of both data and schema are claimed to be the possibility to infer and execute queries which were not initially envisaged.

Integration of two heterogeneous data sources for associating genotype (Gene Ontology) to phenotype (Entrez Gene) information using *RDF* is described in (Sahoo et al., 2007). Rules are used to make associations based upon *isA* and *partOf* relationships. In our case, the relationships are between classes, and also between individuals.

Another very relevant work is introduced by (Obitko, 2007) about the translation of ontologies in *Multi-Agent Systems* in the manufacturing domain. The rules are transported via messages and are interpreted in respective agents. In our opinion, when the rules are executed in sequence then the inferred triples are added to the model which are not necessarily being transported or referred. This might not be a requirement in specific manufacturing application but certainly it is an issue if one has to benefit from the *open world reasoning* provided by ontologies in *DL*.

2.5 Semantic Web Potential for Connecting Knowledge

For connecting knowledge in the development of *Information Systems (IS)*, the role of *Conceptual Schema Centric Development (CSCD)* in comparison with other approaches such as the *Architecture-Centric Development*, the *Test-Driven Development (TDD)*, the *Model-Driven Architecture (MDA)* and the *Domain-Driven Design*, is highlighted as a Grand Challenge for *IS* research (Olivé, 2005). The *CSCD* vision is towards an approach where the schemas are explicit, executable and evolving. The domain knowledge encoded by an *IS* is called the *Conceptual Schema (CS)* which is a combination of *Domain Conceptual Schema (DCS)* and the *Functional Specification (FS)*. The *DCS* describes the domain knowledge independent of the *IS*, whereas the *FS* is about the description of the entity and relationship types of the *DCS*, the related query events and the generating conditions of the queries. The *DCS* developed on these lines correspond closely to ontologies (Olivé, 2004). The *Semantic Web* provides us the technology where the conceptual schemas or ontologies encoded in different possible dialects of *OWL* enriched with the semantics of the domain knowledge, are explicit, executable, and are able to evolve over time. Ontologies, the key element in the *Semantic Web*, permit to explore the information in numerous possible ways (Garcia and Sicilia, 2003). Their exploitation for accessibility would accelerate the inclusion and improve the *eQuality* for people with special needs in the digital world. It is possible to formally specify the concepts in ontologies and hence remove the ambiguities so that the interaction conforms with the needs of the task at hand, and user abilities. The output is presented using additional devices and techniques called the Assistive Technology such as screen reader, electronic Braille, screen magnifier, usage of Alt text and sticky keys, according to the *Web Content Accessibility Guidelines* (WAI, 1997).

Useful Features

The grounding block of Semantic Web, Resource Description Framework(*RDF*) (Beckett and ed., 2004), emphasises on the notion of resources being uniquely identified using Uniform Resource Identifier (*URI*) (Berners-Lee T. and L., 1998). The resources are modeled as triples of the form *Subject, Predicate, Object (SPO).* For an overview, see the *Semantic Web Architecture* in Fig.2.3. It offers the possibility to describe the resources in a commonly agreed upon way globally or within *Communities of Practice.*

Ontologies are developed using RDF Semantics(*RDFS*) (Patrick and Brian, 2004) and Web Ontology language(*OWL*) (Bechhofer et al., 2004) as top level modeling blocks for explicit description of semantics which are machine understandable. The hierarchy of the concepts is established by sub-class relationship and is also directly inferable by the reasoners. More complex relationships are formed by annotating the resources using meaningful domain oriented frame slots.

URIs enable to query multiple and heterogeneous data sources simultaneously using *SPARQL* (Prud'hommeaux and Seaborne, 2007) an *SQL* like query language for *RDF* data. It is not necessary to predetermine the queries. The reasoners enable the query language to explore the data according to active *RDF* model. Based upon the requirements it is possible to formulate a new ontology which is formed by connecting the *URIs* across ontologies. Additionally, it permits exposing *SPARQL* endpoints as resource centered *REST(Representational State Transfer)* style services (Fielding and Taylor, 2002). This way, a part of the coding is performed at a more abstract level and in a more flexible manner than before.

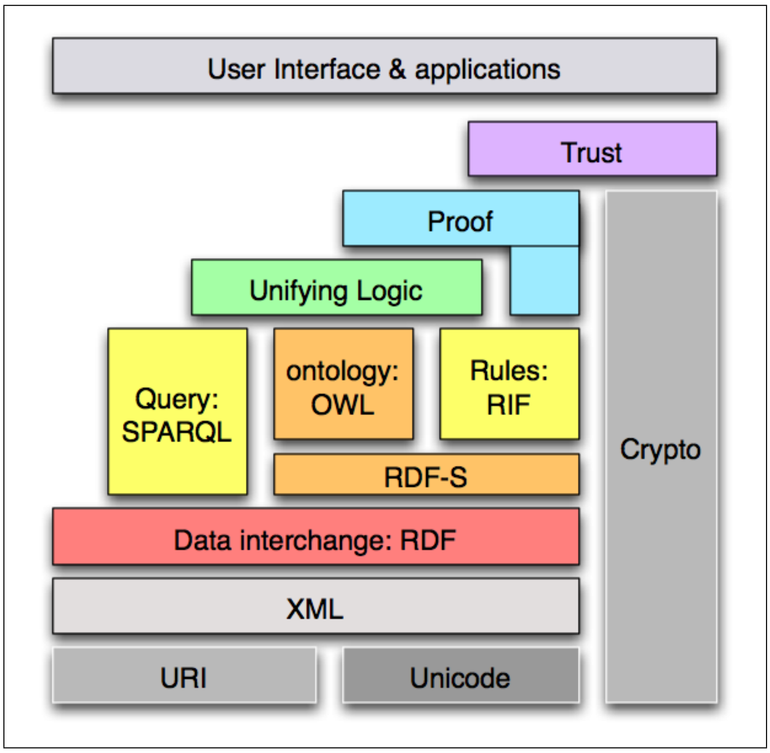

Figure 2.3: Semantic Web Architecture

2.6 Overview of Prototype Testbed

Our prototype and testbed for subsequent enhancements is *SemanticLIFE* (Ahmed et al., 2004), which is a *Personal Information Management System* for managing associations between the user's lifetime information items such as emails, browsed web pages, documents under process and processes running on users computer (see architecture in Fig. 2.4). It has an additional feature to plug-in the *Google Desktop* as another data feed. In other words, it is user's digital diary which is storing the users lifetime electronic activities, not merely the documents. It is a *Java*-based open source framework built into the *Eclipse* plug-in environment. The *Semantic Web* technology is used to transform and store the information items meta data into *Resource Description Framework (RDF)* (Beckett and ed., 2004) triples in accordance with our core ontology consisting of items as classes and their meta data as properties. The queries are either launched by sending *SPARQL* query strings (Prud'hommeaux and Seaborne, 2007) or programmatically using *Jena Ontology API* (Reynolds, 2007). More sophisticated queries are realized by enhancing our ontology to *RDF Semantics (RDFS)* (Patrick and

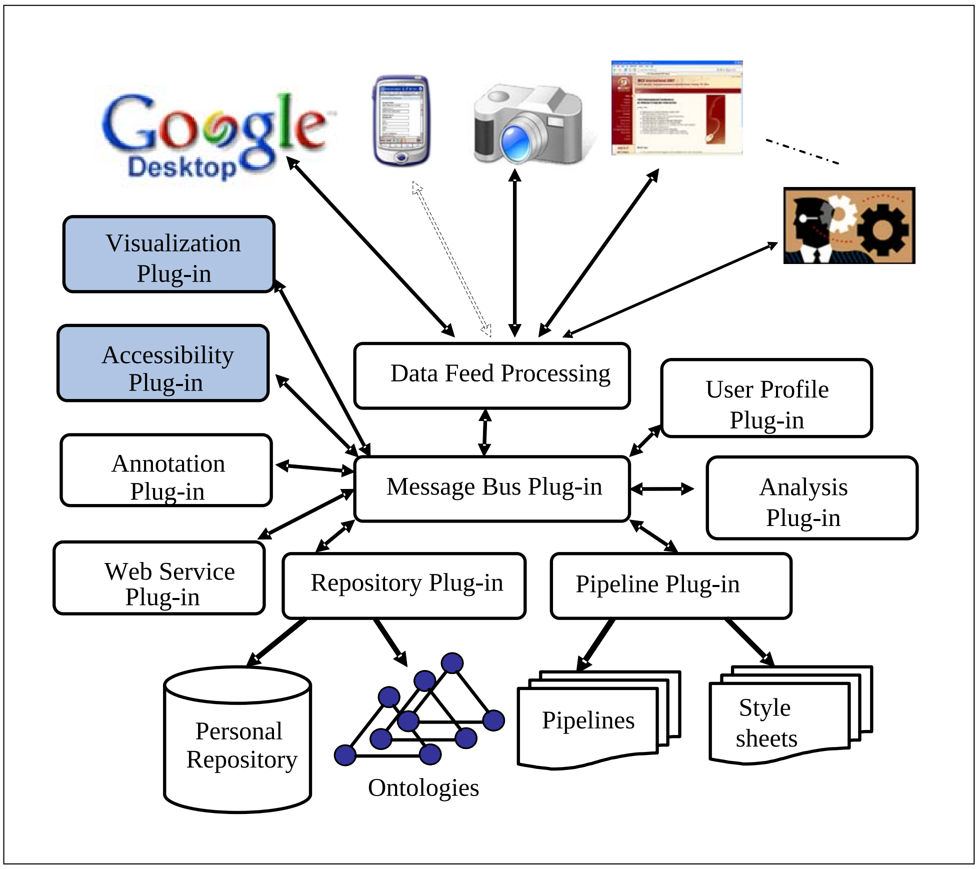

Figure 2.4: SemanticLIFE Plug-in Architecture

Brian, 2004) and applying rules via *Jena Rule Engine*. The brief workflow of the system is as follows:

- *Data Acquisition:* It is accomplished using *"Data Feed Processing Plug-in"*. Data that are fed into the system come from various sources, as mentioned above, and in various formats. Consequently, the acquisition module must be able to handle all these types of data. Instead of constructing a "central" acquisition module with multi source support, several independent modules have been developed, each handling one type of data. Since, the data sources are also vendor specific, therefore this approach was also helpful from maintenance point of view. The acquisition module provides mechanisms to collect information related to the data (meta data) either by automatic extraction based on some predefined structure or by user intervention in form of manual annotation. Automatic extraction at capture time reduces tedious user load, and manual annotation enriches the data with user-defined descriptions which are comparatively easy to recall. The acquisition module wraps data and meta data in an XML based message format and uses *SOAP* (Simple Object Access Protocol) over *http* as transport layer to feed the data items to the *SemanticLIFE* systems *Message Bus Plug-in*. Two types of data acquisition can be distinguished, namely automatic or scheduled feeding and manual feeding. Retrieving email, monitoring user processes (applications) or web-browsing sessions are examples of the former, while manual upload of a file (document, audio, video) and integrating (synchronizing) calendar data to the system falls into the second category. In addition, the *Google Desktop* is incorporated as a separate data feed for the system. Although it is designed as a personal information management system, privacy concerns become an issue as the system tries to capture as much information as possible in an automatic manner. To support a large degree of user control over the feeding process, a range of filtering mechanisms allows to specify which data items are to be forwarded into the system. Examples are the time-based or domain based exclusion of certain web-browsing activities; feeding of email based on sender address or subject fields/keywords; the differentiation between public and private calender entries; filtering of process monitor data based upon specific process, user or application identification. It is important to note that the meta data for each data feed is first converted into *RDF* according to the core ontology for our system and then stored in the repository.

- *Communication Framework:* It consists of three basic plug-ins namely *Message Bus, Web Service and Pipeline* which work in coordination with each other (Anjomshoaa et al., 2006).
  - *Message Bus:* It is responsible for routing and monitoring message traffic, adding time-stamps to messages, and logging system states. This allows analyzing the behavior of the system in case of problems. Moreover, the usage of a message oriented design provides means for future enhancements to guarantee scalability and flexibility. The message, as depicted in Fig. 2.5, has a header and body element.

    The header contains mainly a unique message ID and a system time-stamp at arrival time. Moreover, every message is of a specific type like query message, data feeding message, result message or control message. The message travels between the systems modules. When a module processes the message, a *ProcessComponent* tag is added to the body part of the message to keep trace

```xml
  <?xml version="1.0" encoding="UTF-8" ?>
- <message xmlns="http://www.ifs.tuwien.ac.at/semanticlife#">
  - <header>
      <id>1103774385332</id>
      <timeStamp>Sat Dec. 23 15:38:06 CET2006</timeStamp>
    - <origin>
        <component version="0.3">CalendarFeed</component>
        <timeStamp>Sat Dec. 23 15:38:05 CET2006</timeStamp>
      </origin>
    </header>
  - <body>
    - <messageData contentType="Text/Plain">
        <!-- original immutable message contents -->
      </messageData>
    - <processComponent>
        <component version="0.2">AnalysisModule/Lucene</component>
        <timeStamp>Sat Dec. 23 15:38:06 CET2006</timeStamp>
      - <body contentType="text/xml">
          <!-- result of processing -->
        </body>
      </processComponent>
    </body>
  </message>
```

Figure 2.5: SemanticLIFE Message Structure

of the processing steps. The contents of *ProcessComponent* includes the time stamp and results of the processing. Eventually, the body part contains the original content of the message which could be email, a document, or could be in any multimedia format such as the picture or audio plus the results of the various analysis steps. Binary contents of the body are encoded to *Base64* for keeping them safe from validating and parsing errors.

– *Web Service:* This plug-in provides an interface for our internal services as well as making our system open to external *Web services* using the specifications in respective *WSDL* files. It offers the *"finder service"* for a request, *"invocation service"* for its invocation, and *"recommender service"* to semantically rank the service results.

– *Pipeline:* It plays a key role in the creation and orchestration of business services. *A pipeline* is introduced as *a uniquely named set of service-calls and intermediate transformations*. The pipelines depict different business scenarios and are encoded in *XML*. It is possible to combine different pipelines in various ways and perform required transformations which may vary from typical data transformations such as *HTML* to *RDF*, and also semantic transformations such as enhancing your tourism plan with user preferences based on user profile ontology.

- *Information Analysis:* The basic job is the *XMLization* of the contents and the extraction of meta-data. The message meta data is then transformed into *RDF triples* and forwarded to *Repository Plug-in* for storage. Two kinds of information are considered, namely *message meta-data* and information extracted from the *message content*:
  - *message meta-data:* These are the header fields for any data source. For example, for *email*, the message meta data consists of email header information such as "sender", "receiver", "subject", "date", and for *File Monitoring*, this consists of fields such as "file name", "size", "modification date", "owner". Also, the message meta data consists of some informative fields marking the entry of each message into our system, such as "time stamp", "message id", "component id".
  - *message content:* This is the message body such as contents of the email body and actual file contents. For textual analysis and extraction of meta data, existing solutions like Apache Lucene and *GATE* (Cunningham et al., 2002) are used.
- *Storage:* The storage is carried out by the *Repository Plug-in* which interacts with *ontologies (information schemas)* under local file system, and the *RDF triple store* under *MySQL*. It is designed in a technology independent way so as to cater for different types of storage technologies.
- *Annotation:* It provides the ability to annotate the resources in our repository. Both *"free text annotations"* and *"semantic annotations"* are supported (Latif et al., 2006). In *"free text annotations"*, the user may give comments about any resource as a whole. For example, while browsing a web page the personal remarks about the usefulness or quality of the web page, or while reading a paper the technical remarks may be entered. The *semantic annotations* are made against existing triples about a resource. For example, for an email or a photo there are many triples which are generated against its meta data. For each of these meta data fields the annotation can be made which are later exploitable in context of ontological reasoning.
- *Visualization:* This plug-in provides the *User Interface* for our system, as well as some intuitive visualizations. The *UI* is designed to fulfill

the end user's tasks as detailed in requirements in Section 3.1.1 and 3.1.2.

- *Accessibility:* In oder to adapt the *UI* for people with special needs, a *Generic Accessibility Framework* is proposed which is described in detail in Chapter 5 and Chapter 6. In summary, *User Impairments* and *UI Characteristics* are first modeled and encoded in *OWL-DL* ontologies. Then, the ontological entities of both are connected using *rules* resulting in a new ontology consisting of *RDF Triples* which describe the suitable *UI Characteristics* for specific *User Impairments.* We term it as the *"Connecting Ontology"* between the *User Impairments* and the *UI Characteristics* or *ImpUiCo* in short. The *ImpUiCo* is further processed by the *Visualization* plug-in for adapting the *Style Sheets.*

2.7 Summary

Different aspects of *accessibility* are explored. The literature suggests the concept to be applicable for all, contrary to most of the actual practice. It is observed that there is a need to interconnect the different contextual elements for making significant improvement towards *accessible computer systems.* Therefore, a generic approach will be more cost-effective, popular within the industry, and would be helpful in providing *Acc4All.* Insights from *Data Integration* and *Semantic Web Technology* suggests that it is possible, and could be the approach in the right direction. The status of our prototype system is described which is capable of being used as a testbed for the research made in all its functional components.

Chapter 3

User Interface Design Considerations

A good *User Interface* design is essential for performing the tasks in an optimal way while giving utmost consideration to user's convenience. The end user can conveniently perform the tasks, provided the mental models of both the designer and the user, and the implemented system image are coherent with each other (Dix et al., 1993)(Laurel, 1993). Practically, this triangle is never perfect due to lack of shared semantics at each point. The situation is aggravated in case of users with special needs. For *SemanticLIFE*, an effort was made to gather data related from the intended users, the tasks which could possibly be performed, and an analysis as to how different users would like to carry out those tasks. The intention was to base the *UI* design and the related business components in the light of user requirements, i.e., adapting a user centered approach. The activity was carried out within our group of about 10 researchers. It consisted of individual face to face discussions with group members and joint group discussions in an iterative manner. The methods used were interviewing, designing and filling questionnaires, and making functional prototypes. The prototypes were discussed, and consequently refined in three successive iterations before freezing the agreed upon *UI* for the first version of the system.

As a first step, general *UI standards* were defined. The purpose was to get the suggestions from group members while enforcing homogeneity between *UI* for all the components of the system which were being developed by different colleagues.

3.1 End User Interaction Goals

In order to build a system with good *Usability*, there already exist guidelines in the literature such as (Nielsen, 1994)(Nielsen, 1999)(Holzinger, 2005). We adapted those for our system which is based upon ontologies for managing the information. The input to *SemanticLIFE* is user's lifetime information items captured by independent data feed modules, as described in 2.6. Our primary goal is to explore the associations of information items. The secondary goal is to visualize their contents to explore further associations. The associations are of three types. Firstly, these are based upon the already available meta data of an information item, for example *EXIF Header* fields for pictures, and email header information. Secondly, the annotations given by the user, for example *email1* is related with *file1*. Thirdly, the associations discovered from the contents of the information items, e.g., finding the string *"eclipse svn problems"* in an email body, and associating it with web page *"http://subclipse.tigris.org/"*. In context of these goals, the general user interface requirements and the accessibility requirements are discussed in the following sub-sections.

3.1.1 User Interface Requirements

1. The user interface is to be interactive
2. The user interface should be stable (not too dynamic)
3. Ability to generate query from the same user interface
4. Ability to do annotation on the same user interface
5. Both graphical and textual display
6. Ability to view classes (primitive + user defined) from the ontology
7. Ability to view associated classes, annotations and properties as described in the ontology
8. Ability to see context menu for each selected item
9. Display mechanism for each item specified in ontology, and thus under user control
10. The layout of the display under user control by drag and drop style

3.1.2 Accessibility Requirements

People with special needs are at a disadvantage with others when they are required to interact with the system in varying circumstances. Since it is not a collaborative but a personal system, a personalized *UI* would fulfill the user's requirements better than a universal interface. However, using appropriate style sheets, the *UI* can be adjusted for any user. In general, the *UI* should conform to the accessibility checklist provided at (WAI, 1997). Fortunately, with current technology of hardware and software many things are possible which could improve the interaction for people with special needs. Before going into that, some of the essential aspects to take into consideration are as follows:

Assistive Technology Support

"Assistive Technology" is any product or service that is used to maintain or improve the access to computers for people with disabilities either directly or by means of giving support to caregiver [1]. Some examples are Screen Reader, Electronic Braille, Touch Screen, Switch Activated EADL (Electronic Aids to Daily Living), Keyboard with Large Print Keys, EZKeys for dual word prediction and abbreviation expansion, Screen Magnification, Watches with Alarms, Joystick, Keyboard Alternatives such as Sticky Keys, One Hand Keyboard, On Screen Keyboard and Eye Gaze System.

Multiple Device Support

There should be a possibility to use the system with the help of multiple devices. For example, there are some handicapped persons who can only use Television with a little or no help from the remote control. In this case, it would be recommendable to provide the system interaction via Television instead of trying to train the challenged user for working with the computer.

Caregiver Support

In many cases, the impairment severity of the user renders it impossible to use the system. In these cases, the caregiver of the handicapped person may interact with the system and follow the system generated recommendations. It is to be kept in mind that our system is storing the lifetime information of a person, that can greatly support the user in many scenarios including information searching and recall of the past events. However, in case of

[1] AbilityHub, Assistive Technology Solutions, http://www.abilityhub.com/ (5th September 2007)

interaction by the caregiver, there are security, privacy and legal issues which need careful consideration.

Implications of User's Electronic Activities

Normally, the *Accessibility* is incorporated at the *user Interface*, i.e., the look and feel only. Over the *Web*, it is assumed to be generally applicable for the *Web Contents*. In case of our prototype, the situation is different in the sense that although the user's stored information items do have content part which needs to be made *Accessible*. But also, they represent the user's electronic activities, which at times map to their physical activities. For example, during the HCII 2007 conference visit to Beijing, various telephone calls made, snapshots taken, web pages browsed, files edited and notes taken may represent the following activities:

- Paper presentation
- Opening ceremony
- Great Wall tour
- Research collaboration meeting

The knowledge about the activities is either implicit in the information items, or in user's mind. With the help of meaningful annotations, this implicit knowledge is made explicit and then it can be made *Accessible*. In this way, we can see that using the *Semantic Web Technology*, the *Accessibility* to people with special needs is elevated to another level of knowledge acquisition, i.e, acquiring knowledge about activities and not merely the contents.

We believe that the core information items and the domain concepts are also influenced by user's abilities or disabilities. Consider the scenario of our prototype system which is storing personal photos in the repository. The photos are annotated based upon the picture meta data, and also the contents of the photos. We can say the following:

- $\langle$*Photo relatedWith Place*$\rangle$
- $\langle$*Photo relatedWith Object*$\rangle$

where all the photos in the repository are instances of the concept *Photo*. Moreover, the concept *Place* has instances such as "Restaurant", "Railway Station", "Class Room", and *Object* has instances such as "Fan", "Table", "Plate", "Air Craft". The semantics of the object instances are described separately in the commonly agreed *Foundational Ontologies* within *Communities of Practice*, and these can be accessed by pointing to their corresponding *URI* such as from the OntoWordNet (Gangemi et al., 2003). For example:

"Fan" by "http://www.loa-cnr.it/ontologies/OWN/OWN.owl#FAN";

"Table" by "http://www.loa-cnr.it/ontologies/OWN/OWN.owl#TABLE_1";

"Plate" by "http://www.loa-cnr.it/ontologies/OWN/OWN.owl#PLATE_1";

"Air Craft" by "http://www.loa-cnr.it/ontologies/OWN/OWN.owl#AIRCRAFT";

The entities are described in detail by aligning OntoWordNet terms with the terms in another foundational ontology *DOLCE*(a Descriptive Ontology for Linguistic and Cognitive Engineering) (Ferrario and Oltramari, 2004) (Masolo et al., 2003). As an example, see the step by step description of ".../OWN.owl#FAN" in the listing 3.1.

Listing 3.1: Description of Fan in OntoWordNet aligned with DOLCE

```
<owl:Class rdf:about="..../OWN.owl#FAN">
      <rdfs:comment>a device for creating a current of air by
              movement of a surface or surfaces
      </rdfs:comment>
      <rdfs:subClassOf>
              <owl:Class rdf:about="..../OWN.owl#DEVICE_1"/>
      </rdfs:subClassOf>
</owl:Class>
.....
....
<owl:Class rdf:about="..../OWN.owl#DEVICE_1">
      <rdfs:comment>an instrumentality invented for a particular
       purpose; ....
      </rdfs:comment>
      <rdfs:subClassOf rdf:resource="..../OWN.owl#INSTRUMENTALITY_
INSTRUMENTATION"/>
</owl:Class>
......
....
<owl:Class rdf:about=
              "..../OWN.owl#INSTRUMENTALITY_INSTRUMENTATION">
      <rdfs:comment>an artifact (or system of artifacts) that is
       instrumental in accomplishing some end
      </rdfs:comment>
      <rdfs:subClassOf rdf:resource=
              "..../OWN.owl#ARTIFACT_ARTEFACT"/>
</owl:Class>
......
....
```

```
  <owl:Class rdf:about=".../OWN.owl#ARTIFACT_ARTEFACT">
        <rdfs:comment>a man-made object</rdfs:comment>
        <rdfs:subClassOf>
                <owl:Class rdf:about=
                        ".../OWN.owl#OBJECT_PHYSICAL_OBJECT"/>
        </rdfs:subClassOf>
  </owl:Class>
   ......
   ....
  <owl:Class rdf:about=".../OWN.owl#OBJECT_PHYSICAL_OBJECT">
        <rdfs:comment>a physical (tangible and visible) entity; it
        was full of rackets, balls and other objects
        </rdfs:comment>
        <rdfs:subClassOf>
          <owl:Class rdf:about="http://www.loa-cnr.it/ontologies/
DOLCE-Lite.owl#physical-object"/>
        </rdfs:subClassOf>
  </owl:Class>
   ......
   ....
  <!-- Class: http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl
 #physical-object-->

  <owl:Class rdf:about="#physical-object">
        <rdfs:comment>The main characteristic of physical objects is
        that they are endurants with unity. ......
        </rdfs:comment>
        <rdfs:subClassOf>
                <owl:Class rdf:about="#physical-endurant"/>
        </rdfs:subClassOf>
  </owl:Class>
   ......
   ....
  <!-- Class: http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl
 #physical-endurant-->

  <owl:Class rdf:about="#physical-endurant">
    <rdfs:comment>An endurant having a direct physical (at least
         spatial) quality.
    </rdfs:comment>
    <rdfs:subClassOf>
     <owl:Restriction>
       <owl:onProperty rdf:resource="#has-quality"/>
       <owl:someValuesFrom rdf:resource="#physical-quality"/>
     </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
     <owl:Restriction>
      <owl:onProperty rdf:resource="#has-quality"/>
```

```
      <owl:allValuesFrom rdf:resource="#physical-quality"/>
     </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
     <owl:Restriction>
      <owl:onProperty rdf:resource="#part"/>
      <owl:allValuesFrom rdf:resource="#physical-endurant"/>
     </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
     <owl:Restriction>
      <owl:onProperty rdf:resource="#has-quality"/>
      <owl:someValuesFrom rdf:resource="#spatial-location_q"/>
     </owl:Restriction>
    </rdfs:subClassOf>
   <rdfs:subClassOf>
    <owl:Restriction>
     <owl:onProperty rdf:resource="#specific-constant-constituent"/>
      <owl:allValuesFrom rdf:resource="#physical-endurant"/>
     </owl:Restriction>
   </rdfs:subClassOf>
    <rdfs:subClassOf>
     <owl:Class rdf:about="#endurant"/>
    </rdfs:subClassOf>
  </owl:Class>
   ......
   ....
  <!-- Class: http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl
#endurant -->

<owl:Class rdf:about="#endurant">
 <rdfs:comment>The main characteristic of endurants is that all of
        them are independent essential wholes. .....
 </rdfs:comment>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#part"/>
   <owl:allValuesFrom rdf:resource="#endurant"/>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#specific-constant-constituent"/>
   <owl:allValuesFrom rdf:resource="#endurant"/>
  </owl:Restriction>
 </rdfs:subClassOf>
 <rdfs:subClassOf>
  <owl:Class rdf:about="#spatio-temporal-particular"/>
 </rdfs:subClassOf>
```

```
 <rdfs:subClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="#participant-in"/>
   <owl:someValuesFrom rdf:resource="#perdurant"/>
  </owl:Restriction>
 </rdfs:subClassOf>
</owl:Class>
```

The semantics described at above *URIs* may not be understandable for people with special needs. Especially, when the reasoning is being performed based upon the existing semantics then the search results could make a great difference. Therefore, there is a need to represent them differently which is suitable for the specific impairments. Though, we have not yet solved this problem significantly. However, it can be tackled to some extent while giving annotations to information items and their contents (Karim and Tjoa, 2006a). In this regard, the caregiver of the user with disabilities could be very helpful.

3.2 Analysis of End User Tasks

The next step was to design the questionnaires so as to formally get more requirements related with the end user tasks. Once the tasks are identified, then we can decide where to get those executed, i.e., distributed in *UI* and business components, or entirely in business components. This would be followed by the detailed design. For this purpose, the feedback from the team members were collected (see the questionnaire in Appendix A).

3.3 User Interface Mock-ups

The paper prototypes and non functional screen shots were prepared which were based upon the collected requirements and data about tasks.

3.3.1 User Interface Standards

Initially, we adapted a trivial list of *UI* standards so that the team members are able to design and implement the look and feel of their sub-systems homogeneously. The standards were as follows (refer to Fig. 3.1 for a good comprehension):

1. Main menu bar at the top

2. Status bar at the bottom

3. A tabbed pane (*Eclipse* style tabbed panes) at the left showing the categorized class hierarchy. One of the categories can be for system management covering user profile, query, annotation, ontology management, etc.

4. A tabbed pane at the right for filtering / control options. The default is time based filtering managed by a *GUI* component, text entry fields and horizontal time range sliders.

5. A collection of tabbed panes in upper middle part providing interface for various purposes such as visualization of items, query generation, annotation and user profile management.

   (a) A tabbed pane in lower half of each of the above tabbed pane for visualization of the contents in different formats such as *Text*, *RDF*, *XML*.

   (b) In turn, each of these tabbed panes would have the same default layout (from point 1 to 6)

6. Upon selecting the appropriate operation in main contents visualization pane, the corresponding tab will also become active. For example, right click on a class will display a context menu (generate new query, new annotation, etc.). Selecting one of these options (say "generate new query") will activate the query tab as query *UI*. The changes made on query tab will take effect on main contents visualization. Similar will be the case for other tabs.

   The display mechanism for the above *UI* would also be specified in the ontology. Each item to be displayed (interface component or user data) is of specific type and has its own semantics. These are to be coded in ontology and displayed using the *XSL* transformation.

3.3.2 User Interface Description

For finalizing the prototype *UI*, four iterations were planned. However after three iterations it was agreed to conclude the prototype till implementation. During each iteration the prototype was presented to the group for discussion and refinement. The final version of the prototypes are given in Figure 3.1 and Figure 3.2.

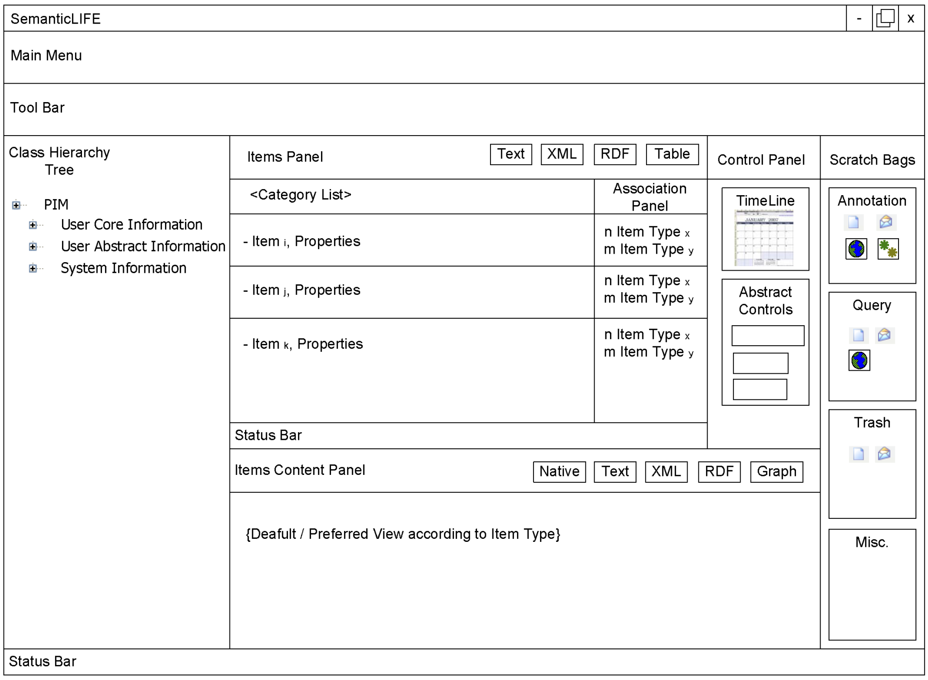

Figure 3.1: Concept of SemanticLIFE UI, the Default View

Behavior of UI Components

Explanation about the significant UI components (see Fig.3.2) is as follows:

1. *System Modes*; There can be different modes in which the user may interact, such as *Query*, *Annotation* and *Trash*. The purpose of providing these modes is to reduce the number of mouse clicks. Also, once a mode is selected all the subsequent clicks would mean the same operation on the clicked entity which is sent to the appropriate *Scratch Bag* for offline handling. For example, when *Query* mode is selected the shape of cursor also changes as shown in the figure. Subsequently, wherever the user moves the cursor and clicks the mouse, the entity under focus is queried immediately, or it is marked for later querying in batch mode and temporarily copied in *Query Scratch Bag*.

2. *Scratch Bags*; The items placed here can be worked upon in offline mode according to the user's convenience. Also, these are used for batch execution of the task under user control.

3. *Class Hierarchy Tree*; The class hierarchy shows the items of our interest coming from the concerned ontologies. The *User Core Information*

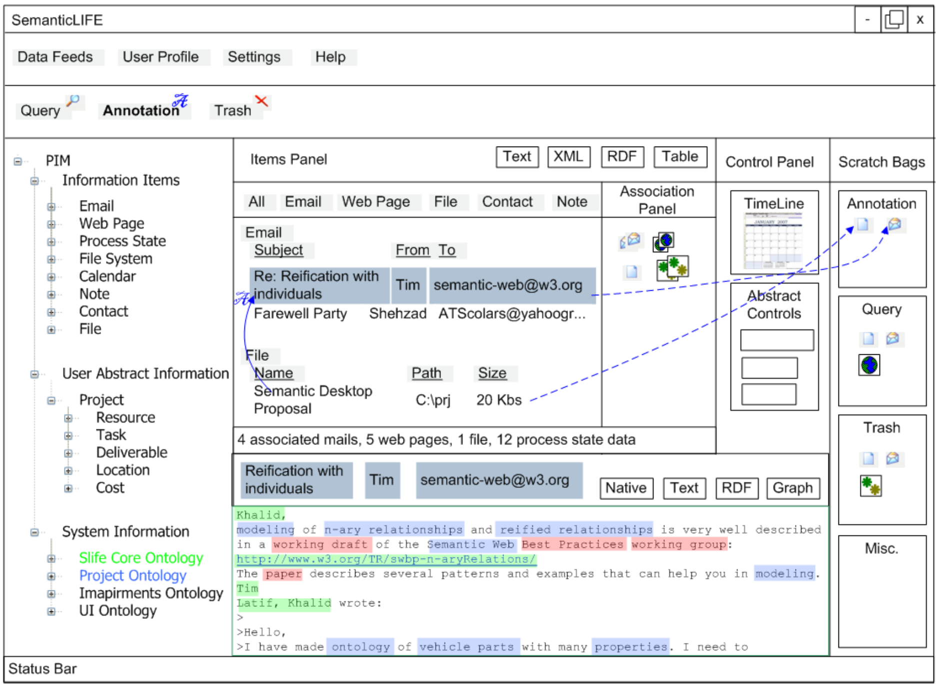

Figure 3.2: Concept of SemanticLIFE UI, the Detailed View With Sample Data

consists of the information items fed into our system by the *Data Feed* plug-in. The *Abstract Information* consists of higher level conceptual information derived from the core user information by applying certain rules which work in coordination with the annotations against the ontology resources. The *System Information* consists system managed resources which are not directly usable. For example the ontologies for user core information items, user impairments, task related ontologies like projects ontology or conference ontology.

4. *Items Panel*; The item selected in the *Class Hierarchy Tree* may consist of many child items. All those child items with header meta data are displayed in the *Items Panel*. In the *Items panel*, user has the choice to display the contents in different formats such as *Text*, *RDF*, *XML*.

5. *Items Content Panel*; When an item is selected in the *Items Panel*, then its detailed contents are displayed here. User again has the choice to display the contents in different formats with an addition to display the contents in the *Native* format of the item. For example, the *Native* format for an email item is like an *Outlook* style, for a web page it is like in a *browser*.

6. *Association Panel*; When an item is selected in the *Items Panel*, then all the items which are associated with it are listed here. The user can click these to explore further. The associations are determined based upon the annotations.

7. *Control Panel*; This is used to filter the items. the default option is time line of information items. There can also be abstract controls such as filtering based upon the ranking criteria of different types. For example, to see the items which are associated with the concept "Asian food", or the "Inexpensive food". The result set in user's mind might be to retrieve the photos, telephone calls which were made (and then captured in the system using Data Feed plug-in) in or around locations for "Asian food", "Inexpensive food".

8. *Colors in Items Content Panel*; The colors used in different parts of the screen are functionally significant. The contents depicting the concepts in ontologies, are colored differently based upon their relevant ontology. This makes it easy for the user to select the *Task or Domain Ontology* first, and then select the concepts from the contents in a semi-automatic way for different operations such as *Query* and *Annotation* which also improves the task execution.

3.4 Summary

Based upon the feedback from the users, we were able to design a *UI* for our system which is closer to the requirements of the user, and the tasks. The *Accessibility Requirements* highlighted the importance of different components involved, and the need to persist the semantic relationships between those for automatic user interface adaptation. It also helped us to synthesize our idea to incorporate ontologies into the system in general, and user interface in particular. The survey also helped us to identify the tasks which should be executed offline and in batch mode. Moreover, the innovation of modes such as *Query* and *Annotation*, were stimulating for the developers to think intuitively for designing and implementing the respective sub-systems so as to satisfy the requirements of the end user.

Chapter 4

Conceptual Ontology Design Patterns for Accessibility

It is essential to conceptualize and identify the objectives, scope and components of the application leading to the development of the related ontologies for the concerned application domain. The recent trend of *Conceptual Ontology Design Patterns* in ontology design patterns research (Gangemi, 2005), is the stimulating factor for making some useful improvisations for our system. The essence is the ontological modeling of the domain that starts with the description of a *Generic Use Case (GUC)* for the system, within the framework of a *reference ontology*, which could be *specialized* or *generalized* by the domain experts. A *GUC* is supposed to represent the domain tasks which can be completed by answering specific *Competency Questions*. The *Competency Questions* are the informal questions which must be answered by the ontology (Jones et al., 1998). The *GUC* thus modeled can be encoded formally as a pattern, called the *Conceptual or Content Ontology Design Patterns (CODeP)*.

Another complicated issue for us was the selection of a *reference ontology*, also called the *domain ontology*. The *reference ontology* is an axiomatic representation of the most top level concepts which are reusable in many application ontologies (Menzel, 2003). The examples of those notions are spatial and temporal concepts, objects, attributes, events, different types of relations such as mereology, containment, membership, connections and branches, and constituents (Winston et al., 1987). For our purpose, we found *DOLCE (a Descriptive Ontology for Linguistic and Cognitive Engineering)* (Masolo et al., 2002) to be suitable because of the fact that it is also in *OWL DL* and contains the top level concepts of space, time, event and location. Our prototype *SemanticLIFE* captures the user's lifetime information items. The lifetime items represent the events and may happen

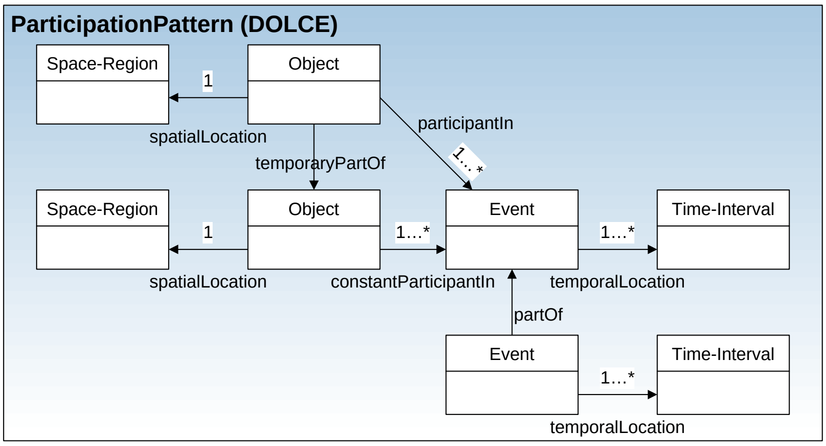

Figure 4.1: Pattern for Participation at Spatio-Temporal Location (DOLCE)

anywhere, anytime. Thus a classical design pattern depicting *participation at spatio-temporal location* (see Fig. 4.1), is inherited and is used in each of our patterns described in next sub sections.

4.1 Generic Accessibility Pattern

A generic accessibility pattern for *SemanticLIFE* (Ahmed et al., 2004) is shown in Fig. 4.2 which is further extend able to cover various scenarios. A few specializations are also shown which are explained later in this section. The description of different components is given below:

- *InformationObject:* These are information items in our system with associated semantics.

- *CausalObject:* It is derived from *InformationObject* and has its own semantics described in an ontology such as Life Events ontology (see Fig. 4.3 for the main concepts).

- *Representation:* The interface or visualization which is used for the visualization of the *CausalObjects*. There can be different visualizations for different users in different contexts and according to user's impairments profile. The semantics of *Representation* are described in terms of their composition, intended user base, data to be shown, and the tasks for which it is designed. For example, tabular representation is usually used for numerical data, and geographical map is more suitable for showing spatially significant information.

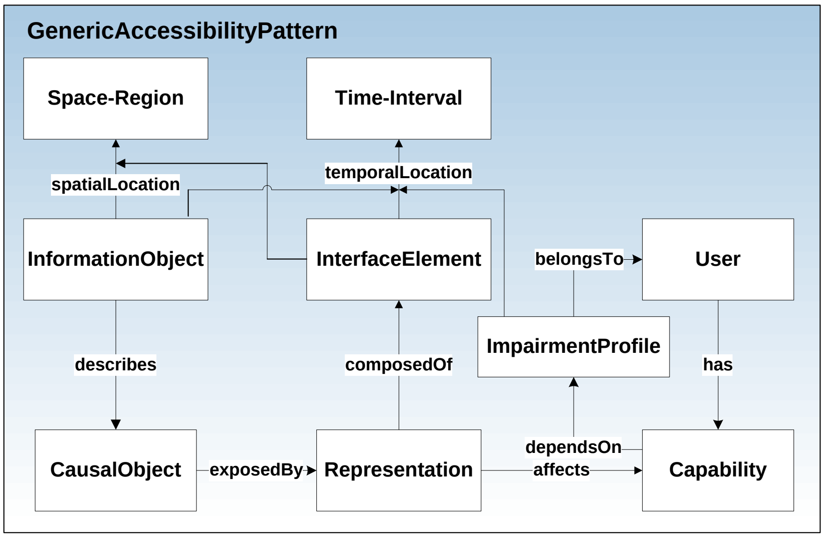

Figure 4.2: Generic Accessibility Pattern with Spatio-Temporal Dimensions

- *InterfaceElements:* Used to make *Representations.* According to the *InformationObject* and *User* profile data, appropriate *InterfaceElements* are selected for composing a *Representation.* Each interface element has associated semantics. The semantics are related with their usability measures and composition with other interface elements to form composite elements.

- *Capability:* It is the ability of the user to carry out a certain task. It depends upon the user's impairments data and the domain oriented task ontology. Based upon the user's task completion statistics the system can be fine-tuned by configuring the different components in the specified pattern.

- *ImpairmentsProfile:* Ontology about user's impairments (disability) related data (Karim and Tjoa, 2006b). The user's *Capability* to carry out certain tasks is dependent upon the related impairment value in the impairments profile.

- *Space-Region:* The concept is inherited from *DOLCE.* For our system, it is possible to trace the position of the information item in space using semi-automatic mechanisms. For example, in case of information items consisting of telephone log, the location of the call can be traced from the prefix of the number which represents the country and city code. Also, the locations may be asserted by annotating the information items. In addition, the concept of spatial location is also

Life Event User's life time events such as birth days of family members and friends, anniversaries, and other occurrences of user's interest
Attributes {Description}

- **Spatial Location** Geographical location of the place where event took place
  Attributes {Name, Coordinates}
- **Temporal Location** Event's starting and ending Date / Time
  Attributes {Start Time, End Time}
- **Disposition** Indicates the nature or mood of the event for the user
  Attributes {Description}

Figure 4.3: User's Lifetime Events

applicable for the presentation of interface elements on specific locations of display devices. Though useful for the system ubiquitousness, we have not deliberately extended the concept to the *User*, because this is not currently foreseen in our prototype. This is achieved indirectly by annotating the information items.

- *Time-Interval:* The concept is inherited from *DOLCE*. Each indicated entity exists in some time duration. Some entities are not linked with *Time-Interval*, but indeed the temporal locations for those are also derivable through the intermediate entities.

A simplified version of the pattern without introducing space and time concepts is shown in Fig. 4.4. In rest of the patterns which follow, we would map the simplified version, for the sake of simplicity only.

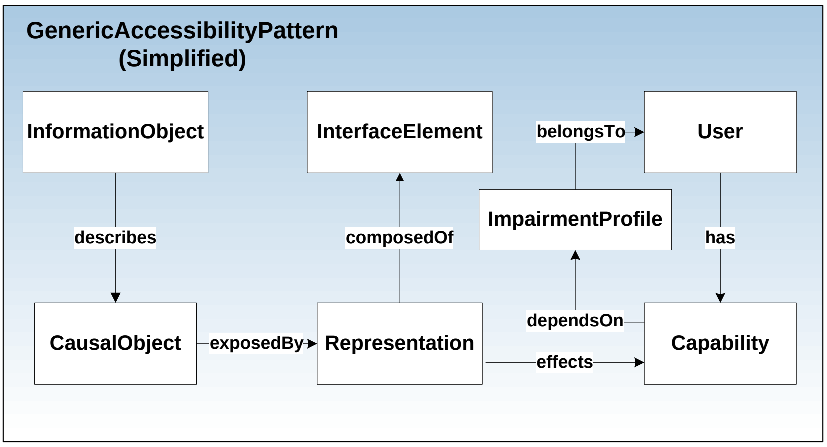

Figure 4.4: Generic Accessibility Pattern (Simplified Version)

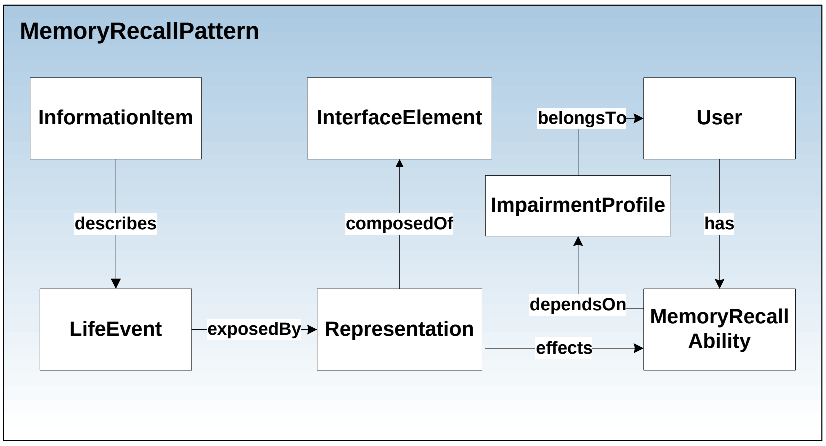

Figure 4.5: Ontology Pattern for Inferring Effect on User's Memory Recall

4.2 Memory Recall Pattern

A specialization of our proposed *Generic Accessibility Pattern* for helping the people with memory deficit is shown in Fig. 4.5. It depicts the relationship between the user's *Ability to Recall* things, the lifetime *InformationItems* stored in user's repository which are associated with *LifeEvents*, and the *Representations* to present or expose those events to the user. In context of our baseline *SemanticLIFE* system the explanation of the pattern is as follows:

- *InformationItem:* Feed items in our system like emails, files of multiple types, web browsing history, chat sessions, processes running on user's PC, contacts and calendar.

- *LifeEvent:* Important events in user's lifetime which might be helpful in recalling other entities and events. Examples of such events are birthdays, anniversaries, other important occurrences. The events are identified and explicitly specified by semi-automatic analysis and annotation of the *InfromationItems*. It is important to keep in mind that there exists an m:n relationship between them.

- *Representation:* The interface or visualization which is used to show these life events to the user. There can be different visualizations for different users because the preferred sequence and identification of the events may vary according to the user's impairments profile.

- *MemoryRecallAbility:* The ability of the user to recall the events or entities such as person and location with the *Representation*. The *Memo-*

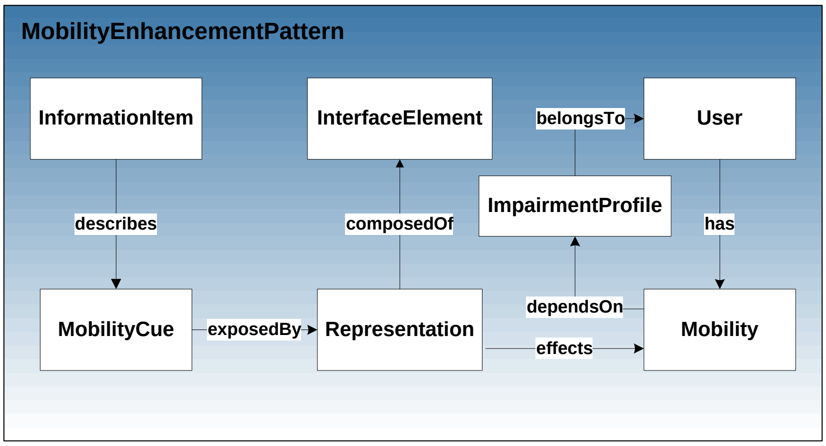

Figure 4.6: Ontology Pattern for Inferring Effect on User's Mobility

ryRecallAbility can be measured against some specific tasks performed by the user designed heuristically.

4.3 Mobility Enhancement Pattern

Another specialization of *Generic Accessibility Pattern* for helping the people with mobility problems is shown in Fig. 4.6. It depicts the relationship between the user's *Mobility Profile*, the lifetime *InformationItems* stored in user's repository which are associated with different *Mobility Cues*, and the *Representations* to present or expose the *InformationItems*, and / or results of any other user queries which might be influenced by user's mobility impairments. The components modified or added for this purpose are described below:

- *Mobility Cue:* These are the indicators for helping the user in locating places or objects with relevant useful attributes for accessibility. For example, a photo of a place can be annotated for describing if the place has accessible ramps, restrooms and lifts. Based upon this information useful suggestions for a trip to already visited and captured places can be generated for a user with mobility impairments.

- *Representation:* The interface or visualization which is used to show the results of user queries. There can be different visualizations for different users because the mobility preferences may vary according to user's impairments profile.

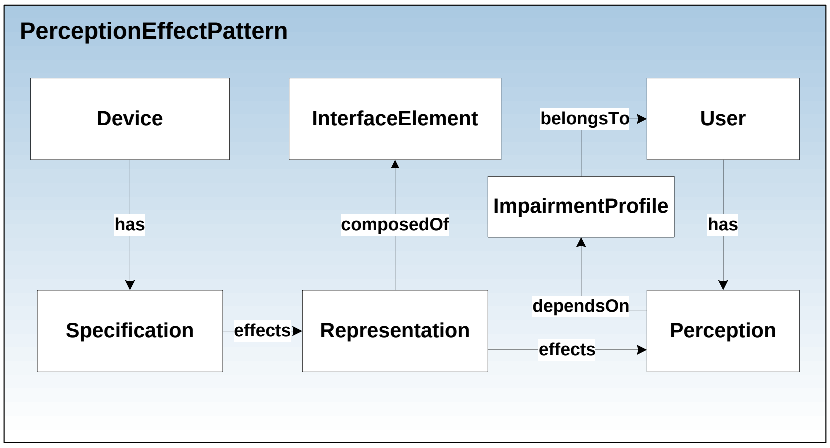

Figure 4.7: Ontology Pattern for Inferring Effect on User's Perception

- *Mobility:* The *Mobility Profile* of the user, part of user's *Impairments Profile*, can be maintained against some specific tasks performed by the user in real life based upon annotation of *InformationItems*.

4.4 Perception Effect Pattern

The *Memory Recall Pattern* is a specialization of the generic accessibility pattern where as the *Perception Effect Pattern* (Fig. 4.7) is its variation. It depicts cause-effect relationship between the user impairments, device profile and interface characteristics. The components modified or added for this purpose are described below:

- *Device:* Interaction device or parts thereof; such as keyboard, display, mobile phone, and PDA.
- *Specification:* This is semantic description of the device, known as the device profile (Klyne et al., 2004)
- *Perception:* User takes time for making sense of the presented information, and then understanding how it is fulfilling the task he / she is destined to do. It can effectively be measured quantitatively and qualitatively when connected with task ontology.

Also, the *CODeP* exercise for *Accessibility* emphasized the need for the following *connecting ontologies*:

- User information Items ↔ Life Events

- Life Events ↔ Representation
- User Impairments ↔ Representation
- Tasks ↔ Representation
- User Impairments ↔ Tasks

4.5 Summary

A generic use case for the system is defined which can be generalized or specialized according to specific scenarios. The above analysis and finalization of accessibility design patterns also helped us to define an *Accessibility Framework* using *Connecting Ontologies* and the related *Services*, which is described in the next chapter. These patterns are to be implemented in *OWL-DL* to enrich the *Connecting Ontologies* for the domains of user impairments, user interface characteristics, specific visualization techniques, domain related tasks, information items and events.

Chapter 5

The *Connecting Ontology*

In order to improve universal accessibility the semantics of the contextual components must be exploited to produce an accessible presentation for the end user. In absence of a generic approach, the provision of universally accessible software is a precarious task for the producers. There are many contributing components interacting with each other (described in detail in Section 2.2 and 2.3). Exploiting enabling technology of *Semantic Web*, we developed an *Accessibility Framework* to combine the different *Contextual Components*. The approach is for developing the *"Connecting Ontologies"* for these components (Karim et al., 2007) capable of giving *suggestions* to suitably match attributes of one contextual component with another. We first introduced the concept of *Connecting Ontology* in (Karim and Tjoa, 2006b) which describes the manual connections between user interface characteristics and the user's impairments data using *OWL-DL*. Now the steps leading to the development of our approach in a semi automatic way, are explained here. First, a description about the concept of *Connecting Ontology* and how it is different from other apparently similar concepts like ontology mapping and ontology integration is given. This is followed by the description of various approaches to explore the connections and their semantic representation. Then the architecture of *Accessibility Framework* is explained which is used for implementation of the proposed approach.

5.1 Overview

An ontology formally describes the concepts in the domain of discourse (Gruber, 1995), or more realistically speaking, helps to formally specify the concepts. A System which is by definition an integrated whole, is essentially composed of heterogeneous components which have to interact with each other to achieve a certain goal. Each of the components can be semantically represented as an ontology, which then raises the challenging issue of connecting these ontologies together pertaining to heterogeneous domains. The *"Connecting Ontology"* $\mathcal{O}_c$ links two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ from *"heterogeneous domains"*, or in other words, describes the linking of *"heterogeneous concepts"* between two such ontologies. This is analogous to the famous wine and food ontology described in (Smith et al., 2004).

For further explanation, let us assume that ontology $\mathcal{O}_1$ has a concept $\mathbf{c}_1$, and ontology $\mathcal{O}_2$ has another concept $\mathbf{c}_2$. The Connecting Ontology $\mathcal{O}_c$ will contain either the concept $\mathbf{c}_3$ to link $\mathbf{c}_1$ with $\mathbf{c}_2$ or extend $\mathbf{c}_1$ with new features which in turn links it with $\mathbf{c}_2$. Once the ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ are populated with instances, their connections are dynamically generated by the reasoner based on their connections present in $\mathcal{O}_c$. It is often the case that the two ontologies are developed based upon different formalisms ensuring decidability of axioms. Although components of both the ontologies are decidable, but when combined together then the decidability is no more guaranteed. In this regard, a useful work is done by (Kutz et al., 2004) describing an *E-Connection* method to link two ontologies in terms of *Abstract Description Systems (ADS)* (Baader et al., 2002).

It is to be kept in mind that we are using the term *Connecting Ontology* in a very much different sense than the other popular notions such as ontology matching, mapping, alignment, articulation, merging or integration [1]. The term *Ontology Mapping* is used in a very broad sense. It is described as "the task of relating the vocabularies of two ontologies that share the same domain of discourse in such a way that the mathematical structure of ontological signatures and their intended interpretations, as specified by the ontological axioms, are respected" (Kalfoglou and Schorlemmer, 2003). The need to map two or more ontologies arise due to simultaneous evolution of many ontologies for the same domain of discourse. This results in terminological as well as semantical differences for the same entities. In this situation when systems interact with each other then these differences must first be resolved. This is one of the major concerns for *Ontology Mapping*.

Ontology Alignment is concerned with the process where binary relations between vocabularies of two ontologies belonging to the same domain of discourse are established. The vocabulary of the ontology specifically means the concepts or unary predicates and the binary relations are the roles such as properties which relate two concepts together. When these binary relations are specified in terms of an ontology in itself, then it is termed as the *Articulation of two Ontologies*. A properly specified articulation ontology helps

[1] http://www.ontologymatching.org/ (5th September 2007)

in *Merging or Fusion* of the concerned ontologies. *Ontology Merging or Fusion* is closer to our notion of *Connecting Ontologies*. However, contrary to fusion, the two ontologies are not merged but connected using additional features and relations. The aspects of composition of ontologies to build a new unified ontology, extending the existing ontologies to build new ones, and incorporating ontologies into the applications are described under *Ontology Integration*.

Interesting insights are observable if we try to look at a representative example in reverse order. Consider the famous food and wine ontology which suggests the best combination of different foods with different wines. Not every type of wine goes well with every type of food. If we explore it further, then it looks logical that the main semantics are related with the taste, the appetite or the stimulation to eat and the digestion. Thinking back-wards, we can conceptualize independent ontologies for food and wine. The food ontology may contain a taxonomy of the food, the calories with respect to the quantity, the tendency to get digested on its own (Quick duration, Medium duration, Long duration), and some other concepts. The wine ontology may also contain a taxonomy of different wines with respect to the costs and some of the concepts such as the appetizing effects (Nil, Low, Medium, High) and the digestive effects (Nil, Low, Medium, High). Now the $\mathcal{O}_c$ can be planned with *Competency Questions ($\mathcal{CQ}$)* such as the following:

Ordering food and wine with a wider variety of combinations like;

- Food requiring no drinks
- Food requiring drinks of certain quality and brand
- Food and wine combination requiring minimum time to digest

The basic level knowledge indicated above is generally available and it maybe obtained from the food and wine producers as structured or unstructured documents. By persisting the domain experts' knowledge as explicit rules, the reliance on them can be reduced considerably.

Another example (see Table 5.1) is about suggesting suitable user interface components according to user's impairments profile. This is often hard coded into the system.

A synthesis of the table data reveals the conceptual relationship between the right and left column which is shown as an *RDF Graph* in Fig. 5.1.

By defining the discrete measurement scales for both impairments and the user interface convenience measures, we are able to state the two relationships as follows:

Table 5.1: Matching Impairments with User Interface Components

Impairment	*Related With*	*UI Component*
NormalMotorControl	*suggests*	ComboBox
NormalMotorControl	*suggests*	RadioButton
NormalMotorControl	*suggests*	CheckBox
NormalMotorControl	*suggests*	ScrollBar
NormalMotorControl	*suggests*	Spinner
NormalMotorControl	*suggests*	ToggleButton
VeryLowMotorControl	*suggests*	AudioFeedback
VeryLowMotorControl	*suggests*	ToggleButton
VeryLowMotorControl	*prohibits*	Spinner
VeryLowMotorControl	*prohibits*	ScrollBar

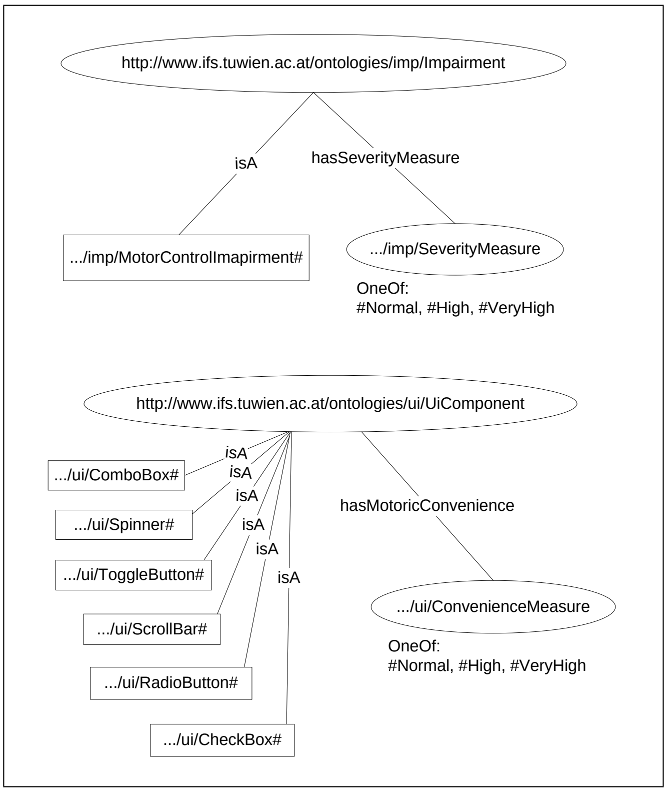

Figure 5.1: Exploring Relationship Between Motor Control Impairment and UI Components

- $MotorControlImpairment$ with $SeverityMeasure$ VeryHigh $suggests$ $UiComponent$ with $MotoricConvenienceMeasure$ "VeryHigh"

- $MotorControlImpairment$ with $SeverityMeasure$ Normal $suggests$ $UiComponent$ with $MotoricConvenienceMeasure$ "Normal"

  In *DL*, these can be written as:

- *VeryLowMotorControl* $\subseteq$ *MotorControlImapirment* *VeryLowMotorControl* $\equiv \exists$ *hasSeverityMeasure*($\ni$ *severityMeasure*.{VeryHigh})

- *NormalMotorControl* $\subseteq$ *MotorControlImapirment* *NormalMotorControl* $\equiv \exists$ *hasSeverityMeasure*($\ni$ *severityMeasure*.{Normal})

  and also;

- *VeryHighMotoricConvenientComponent* $\subseteq$ *UiComponent* *VeryHighMotoricConvenientComponent* $\equiv \exists$ *hasMotoricConvenience*($\ni$ *convenienceMeasure*.{VeryHigh})

- *NormalMotoricConvenientComponent* $\subseteq$ *UiComponent* *NormalMotoricConvenientComponent* $\equiv \exists$ *hasMotoricConvenience*($\ni$ *convenienceMeasure*.{Normal})

The fact indicated by these examples is the following:

The rules describing the connections must first be "accessed", and then those are to be "represented formally"

5.2 Characteristics and benefits

We now describe some of the characteristic features of the *Connecting Ontology* as follows:

- The ontologies to be connected are not related with the same domain of discourse. Though they may be part of the overall application.

- Their vocabularies are independent of each other. Even if there are apparent similarities, they are still assumed to be independent.

- Each of the ontologies to be connected is supposedly developed using its own *CODeP* - explained in Chapter 4.
- There can be similarity between the *CODeP* for two ontologies which may prove helpful for connecting them together.
- Connecting ontologies from heterogeneous domains is in fact creating new knowledge in the light of user's experience, whereas other notions of ontology mapping help to reorganize the existing knowledge.

Following are some of their benefits:

- Useful for top-down evolution of ontologies. Users think in an application or need oriented way, find the available artifacts and then try to connect them together to fulfill their needs.
- The incompatibilities between two ontologies are solved at the ontological level without delving into the application code.
- By elevating one ontology with rich design patterns based upon new user needs, it is possible to trace the corresponding effects on the other ontology due to already established cause-effect relationship between them.
- Viewing another way, the connecting ontologies are in fact helping to automate the coding process.
- Reliance on domain experts reduced due to formal representation of domain knowledge

5.3 Availability of Application Domain Knowledge

If the domain knowledge exists as structured documents then it can be represented formally for automation. Normally this is not the case, and the acquisition of knowledge about the application domain, i.e., the knowledge about the $\mathcal{O}_c$ to be developed, poses serious problems. There can be more than one situations such as the following.

5.3.1 Published Knowledge

The domain knowledge exists already and is available in one of the following forms:

- Ontologies
- Structured Data
- Unstructured Data

Sometimes, the domain is already modeled as *Ontologies*. However, it is not necessary that the available *Ontology* exists in *OWL DL* format. For our method which we describe later, the ontologies in *OWL DL* are required. There are a variety of formats in which an ontology may exist, such as *Topic Maps*, *OWL*, *RDFS*, and *OBO* specifically for ontologies in medical domain. Therefore, the transformation routines must be written or reused if already available, for conversion to *OWL DL*. The conversion from one formalism to another such as from *Topic Maps* to *OWL*, or from one dialect to another such as from *OWL Full* to *OWL DL* is non-trivial and outside the scope of our thesis. However, the recent related initiatives suggests a standard format, called the *Structured Ontology Format (SOF)* which conforms to *OWL 1.1* (Horrocks and Patel-Schneider, 2006). Also, the mapping from *OBO* to *OWL* is described in (Shearer, 2007). In case the ontology exists in *OWL DL*, then the task is shortened, and we can proceed with the next steps.

Sometimes, the domain data is available as *structured documents* like in legacy databases, or formatted files. In this case, the relationship between the data entities is available to some extent. Based upon the available relationships, the schemas can be converted to ontologies, and transformation routines can be used to convert the data to *RDF* triples. Another option is getting the result set by sending queries to external data sources. The conversion of *RDF queries* to *Database queries* is accomplished by using the conversion softwares such as (Bizer et al., 2006).

Also, the published domain knowledge can be found in the form of *unstructured documents* such as web site contents or in a pdf file. In this case, by applying traditional information extraction techniques such as those provided by the frameworks *Lixto* (Baumgartner et al., 2001) and *Gate* (Cunningham et al., 2002), useful data sets can be obtained. Here, the domain knowledge does not exist as a formal ontology yet. The information extraction should be aimed at finding and separating the $\mathcal{O}_1$ related contents, termed as *Ontological Entities* ($\mathcal{OE}$), with those of $\mathcal{O}_2$. Examples of $\mathcal{OE}$ are class names, properties and restrictions. Some sample scenarios are:

- In context of *SemanticLIFE* system; when a user is working on multiple projects at the same time then the information items, which also depict the user activities, compose the probable application domain. The two participating ontologies can be information items ontology (in action as part of *Personal Information Management System* such as *Google Desktop*, *SemanticLIFE*), and the projects ontology (in action as part of project monitoring system being used like *MS Project*, *SVN repository*). The two can be combined to find out the relationship between the activities and the specific entities of the project the user has been working upon.
- Study program recommendations which are based upon the courses offered and the available program choices
- Restaurant menus
- Meteorological warnings and reports, which tell on one hand the prevailing status of the sea and on the other hand the sailing advice based upon the vessel's specifications

5.3.2 Tacit Knowledge

In this case, the application domain knowledge about the $\mathcal{O}_c$, is neither published nor exists in explicit form anywhere. It is a kind of tacit knowledge with domain experts (Nonaka and Takeuchi, 1995). Some relevant examples are:

- The *UI* widgets are chosen and adapted keeping in view the impairments of the user - It is often hard coded, because the knowledge about the semantics does not exist in explicit form
- Suggesting suitable dress, outdoor activity for the prevalent weather forecast

In case of *SemanticLIFE*, the domains of information items and events can be combined to produce a useful $\mathcal{O}_c$, such as the following:

- Match "$\mathcal{O}_1$:*UI* with $\mathcal{O}_2$:*Impairments*"
- Match "$\mathcal{O}_1$:*Dress* with $\mathcal{O}_2$:*Weather*"
- Match "$\mathcal{O}_1$:*OutdoorActivity* with $\mathcal{O}_2$:*Weather*"
- Match "$\mathcal{O}_1$:*InformationItem* with $\mathcal{O}_2$:*Events*"

This would help in answering queries such as *"Should I go skiing tomorrow?"*, or *"Which outdoor activity is recommended in rain?"*.

5.3.3 Undiscovered Knowledge

This is the case where the application domain does not yet exist. Interesting possibilities can be explored in this direction. However, we do not address this aspect in our thesis.

5.4 Extraction of Domain Knowledge for Making Connections

As described earlier, the concept of *Connecting Ontology* ($\mathcal{O}_c$) is different from mapping or merging in the sense that in the later the connection between the ontological entities ($\mathcal{OE}$) is the target between the ontologies from *similar domains*, while in the former the $\mathcal{OE}$ are to be connected between the ontologies from *heterogeneous domains*. Hence, the mapping strategies which use a multitude of similarity measure calculation techniques can not be employed as such for our concept of *Connecting Ontology*.

The manual connection between the two ontologies is the starting point. The *Conceptual Ontology Design Patterns* (explained in Chapter 4) possess vital information which could play a pivotal role in the generation of the information required for making this process largely automatic. This information could serve as a feed for the active engine making the connection between entities. The feed is to be transformed into rules, usable for automating the connection process.

The main concern for developing an $\mathcal{O}_c$ is to formulate the *Competency Questions ($\mathcal{CQ}$)* or scenarios. At an abstract level the $\mathcal{O}_c$ will be performing some tasks in combination with the participating ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$. We assume that $\mathcal{O}_1$ and $\mathcal{O}_2$ already exist in *OWL DL* with well defined $\mathcal{CQ}$ and end user scenarios. The description of *well defined* $\mathcal{CQ}$ in a standardized way is a challenging task. Some people may define the questions as too *vague* covering a large part of the domain but nearly impossible to realise systematically. Whereas, others could define the questions as too *narrow* making it insufficient to fulfill many practical queries. Examples of some $\mathcal{CQ}$ in context of *Impairments Ontology* are:

- Based upon input parameters, determine the impairment and its severity - *too vague*
- Given impairment name and its severity, determine the affected capabilities and their corresponding capability measures - *well defined*
- Given input for perception cues with their respective perception measures, determine the effect on capability - *a little vague*

- Given impairment name, find the concerned organ name - *well defined*
- Given capability, determine the related impairment(s) - *well defined*
- By changing capability measure for one capability, determine the effect on some other capability - *too vague*

The approaches for extracting knowledge about $\mathcal{O}_c$ under our consideration are discussed in the following sub sections. The first one makes use of text processing on the application domain of $\mathcal{O}_c$, and the participating ontologies $\mathcal{O}_i$. The second approach makes use of the standard upper and foundational ontologies for preprocessing of participating ontologies $\mathcal{O}_i$ before making connections. Yet another approach is based upon possible queries (*SPARQL*) on the ontologies $\mathcal{O}_i$.

5.4.1 Using Text Processing Techniques

In this approach, first the published contents of the application domain are matched against the two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$.

1. Parse the contents of available information about the application domain
2. Store the tokens according to schemes offered by these tools
3. Parse the ontology $\mathcal{O}_1$ and store the tokens
4. Compare the tokens (keywords) of application domain with those of $\mathcal{O}_1$
5. Repeat above two steps for ontology $\mathcal{O}_2$

   At this stage, there will be the following three term sets for the application domain:

   - Terms $\in \mathcal{O}_1$
   - Terms $\in \mathcal{O}_2$
   - Terms $\notin \mathcal{O}_1$ or $\mathcal{O}_2$

6. Determine the correspondence of extracted terms from application domain within $\mathcal{O}_i$ tree with respect to $\mathcal{OE}$. This step is only possible if the nomenclature of $\mathcal{O}_i$ corresponds to tokens derived from the application domain, which is mostly not the case. A way out for this problem could be by taking into account the instances which use the

same nomenclature as is used for application domain tokens. The decision about which tokens would serve as classes, properties and slots and which would serve as instances is very important.

7. In the present case of application domain with available domain knowledge, it is perfectly possible to frame sets of $\mathcal{CQ}$. To start with, the typical scenarios would be of type "Connect $\mathcal{O}_1$:$\mathbf{Term}_i$ with $\mathcal{O}_2$:$\mathbf{Term}_j$".

The quality of $\mathcal{O}_c$ thus developed, will be dependent upon the quality of $\mathcal{O}_1$ and $\mathcal{O}_2$. The building mechanism for the $\mathcal{O}_c$ should be executed again upon each new iteration of $\mathcal{O}_1$ and / or $\mathcal{O}_2$. However, adding instances to $\mathcal{O}_i$ should not be a problem, and is in fact one of the goals of $\mathcal{O}_c$.

5.4.2 Using Ontology Alignment

There are some practical problems with the previous approach because of the fact that both the ontologies, though assumed to be made in *OWL DL*, may still have a lot of bottlenecks. Everyone can model an ontology according to his / her understanding and constraints. The differences may culminate in having more than one views about the sets of $\mathcal{OE}$. The solution is to follow the best practices for the ontology development. The best practices must specify for any domain D, the rules and criteria to specify the $\mathcal{OE}$ such as classes, properties, individuals and the constraints. One such effort under research and development is *DynamOnt* (DynamOnt, 2007).

1. The concepts in $\mathcal{O}_1$ and $\mathcal{O}_2$ are matched with concepts in foundational ontologies such as *Suggested Upper Merged Ontology (SUMO)* (SUOWG, 2003), *General knowledge base and commonsense reasoning engine (OpenCyc)* (Reed and Lenat, 2002) and *Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)* (Masolo et al., 2002). After proper alignment the lattices thus formed, $\mathcal{L}_1$ and $\mathcal{L}_2$ for $\mathcal{O}_1$ and $\mathcal{O}_2$ respectively, could be viewed peer to peer. This approach involves extensive user interaction.

2. Now the $\mathcal{CQ}$ should be used by asking which $\mathcal{OE}$ in $\mathcal{L}_1$ may connect with which $\mathcal{OE}$ in $\mathcal{L}_2$. For testing purposes, two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ having correct positional usage of $\mathcal{OE}$ according to agreed upon standard ontologies, can be assumed.

3. A simple case of deriving a class for $\mathcal{O}_c$ from two primitive classes (one from $\mathcal{O}_1$ and the other from $\mathcal{O}_2$) is tried by using *DL* statement and / or Rules.

This approach is promising in the sense that the commonly agreed upon ontologies are being used for the term alignment which would reduce the long term management of $\mathcal{O}_c$ thus developed.

5.4.3 By Exploiting Queries

Since ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ are supposed to exist already, therefore it is possible to frame the sets of $\mathcal{CQ}$ which are the basis for these ontologies. Also, it should be possible to think of all the possible *SPARQL* queries which can be executed to answer these $\mathcal{CQ}$. The parameters used in these query strings may point to $\mathcal{OE}$ about $\mathcal{O}_i$ which can be useful for our purpose. The workflow is as follows:

1. List $\mathcal{CQ}$ for $\mathcal{O}_1$ for the end user scenario(s)
2. For each $\mathcal{CQ}_i$, list the formal queries
3. For each $\mathcal{CQ}_i$, list the $\mathcal{OE}$ involved
4. Repeat above steps for ontology $\mathcal{O}_2$

   The result would be the lists of $\mathcal{OE}$ belonging to $\mathcal{O}_1$ and $\mathcal{O}_2$.
5. In the light of $\mathcal{CQ}$ for the $\mathcal{O}_c$, formulate formal queries
6. In the light of $\mathcal{CQ}$ for the $\mathcal{O}_c$, associate $\mathcal{OE}$ of $\mathcal{O}_1$ with $\mathcal{OE}$ of $\mathcal{O}_2$
7. The above step will be done based upon some rules. The rules can be formulated first in plain language, and then either in a rule language or in *DL*.

Every information space may be connected or required to be connected with another information space to fulfil certain goals. For that, each of these information spaces, formally represented as ontology, has to open its offerings to the outside world. Moreover, the new information space, the *Connecting Ontology*, must also describe its potential offerings. It should also describe the user agent's preferences, and the constraints to help in forming a better connection.

Consider the example of items and packing material. Packing for an item is chosen based upon different factors such as item's dimensions, weight, fragility. Similarly, the impairment ontology represent the user's interaction constraints. These constraints as well as the user's capability profile will have an impact on the user interface components to be used.

Another option is to exploit the result sets of the queries using available *Data Mining* techniques.

5.5 Representation of Connections using Semantic Web Rules

The knowledge about the *Connecting Ontology* $\mathcal{O}_c$ is basically the set of rules which describes the connections from one ontology to another. Once the knowledge about these rules is captured, then the next step is their fromal representation. We chose the *Semantic Web Rule Language* for describing the *"antecedent consequent"* style rules. As is evident from the *Semantic Web Architecture* in Fig. 2.3, the upper layers are composed of rules. That means the rules must be utilized on top of core *RDF* ontologies. The vision is to develop sharable ontologies which will also be in multiple layers (most probably both horizontal and vertical), not just a few huge ontologies from core user information items to user interfaces. In that way, our approach of using rules is perfectly in line with the architectural vision of the *Semantic Web*.

Based upon the description of *Resources* in *OWL*, we can describe the user scenarios in terms of *rules* (Horrocks et al., 2004). The implementation of the rule layer in *Semantic Web Architecture* is still not mature. Some other relevant initiatives in progress are (Battle et al., 2005) (Angele et al., 2005) and (Kaon2, 2007). The *rule* specifying an implication consists of a *body* called the *antecedent*, and a *head* called the *consequent*. The *body* part implies the *head* part, which means that whenever the conditions specified in the *antecedent* part holds then the *consequent* part also holds. Both *antecedent* and *consequent* consist of zero or more *atoms*. An *atom* is composed of *predicates* of arity ≥ 1. The *predicate* is property or relationship expressed in terms of certain restrictions such as *hasUncle* for describing the relationship in a family tree.

There are generally two kinds of reasoning mechanisms used by the *Reasoners* like *Pellet*, *Racer*, *Jena Rule Engine* namely *Forward Chaining Rules Execution* and *Backward Chaining Rules Execution*.

Table 5.2: Sample Triples from Impairments Ontology

Subject	*Predicate*	*Object*
Shuaib	$hasVisualPerception$	Low
Khalid	$hasVisualPerception$	High
George	$hasVisualPerception$	Low

Table 5.3: Sample Triples from User Interface Ontology

Subject	*Predicate*	*Object*
CheckBox	$hasVisualFeedback$	High
Spinner	$hasVisualFeedback$	Low
CascadedMenu	$hasVisualFeedback$	Low

5.5.1 Forward Chaining Rules Execution

It is also called the *data driven* mechanism for reaching the goal which is not specified in advance. Only the conditional statements in the *body* or the *antecedent* part are specified. A search is made of the knowledge base, and if the statements are found then the facts in the *head* or the *consequent* part of the rule are also added to the knowledge base. If another rule is specified with the first one, then the execution for the second rule continues in the above mentioned way with updated knowledge base due to previous rule execution. This mechanism is good for finding the problem when the symptoms are known. Consider the following example:

"Based upon the known impairment type and severity, determine the suitable user interface components". A more specific description can be to *"Connect the user interface components having a specific value for visual feedback, with the specific user's impairment having a specific value for visual acuity"*.

The triples for the two sample ontologies (*Impairments* and *User Interface* are given in Table 5.2, and Table 5.3

- User interface component *"uic"* (*Subject* in Table 5.3) has visual feedback *"vf"* (*Predicate* in Table 5.3) having value "Low" (*Object* in Table 5.3)

Table 5.4: Inferred Model Triples after Rule Execution

Subject	*Predicate*	*Object*
Shuaib	$hasVisualPerception$	Low
Khalid	$hasVisualPerception$	High
George	$hasVisualPerception$	Low
CheckBox	$hasVisualFeedback$	High
Spinner	$hasVisualFeedback$	Low
CascadedMenu	$hasVisualFeedback$	Low
Spinner	$notSuitableFor$	Shuaib
CascadedMenu	$notSuitableFor$	Shuaib
Spinner	$notSuitableFor$	George
CascadedMenu	$notSuitableFor$	George

- User *u"* (*Subject* in Table 5.2) has visual acuity *"va"* (*Predicate* in Table 5.2) having value "Low" (*Object* in Table 5.2)
- Then *"uic"* is not suitable for *"u"*

Now we can write the rule as:

(?uic ui:hasVisualFeedback ui:Low) (?u imp:hasVisualPerception imp:Low) → (?uic eg:notSuitableFor ?u)

where "imp" is the namespace "http://wwww.ifs.tuwieen.ac.at/ontologies/imp#" for the *Impairments Ontology*, "ui" is the namespace ".../ui#" for the *User Interface Ontology*, and "eg" is the namespace for the inferred triples.

When the reasoner is invoked on the base model, which is a union of the *Impairments Ontology* and the *User Interface Ontology*, then an inferred model is created. The inferred model includes the triples from the base model as well as the triples inferred because of rule execution. The actual generated model is rather verbose and contain many *OWL* and *RDFS* elements which are difficult to read. For getting an overview of the mechanism, the triples of our interest can be seen in Table 5.4.

Rules can also be chained together to form a sequence of rules. In this case, care must be taken to keep in mind the newly added triples and their effects on the subsequent rules. In a way, this aspect is useful in specific applications where on the fly actions are required.

5.5.2 Backward Chaining Rules Execution

Contrary to *Forward Chaining*, this is *goal oriented* mechanism. The statements mentioned in the *consequent or head* of the rule, considered as the goal, are processed by the reasoner to retrieve the related statements mentioned in the *antecedent or body*. If the related *antecedent* statements are not found then those are also added as part of the goal. The process goes on until the lowest level goals are satisfied which in turn would help for satisfying the chain of other goals. This method is useful for specific applications where the problem is known in advance and we want to interrogate the reasons behind it. In our case, an example could be the following:

"Based upon the user interface components in use, determine the prevalent user's impairments with severity".

We skip the elicitation of this example for the sake of brevity. Because, detecting user's impairments data based upon the *UI* in use, is not our focus at present. However, it can be back tracked on similar lines as of previous example.

The *cause-effect* relationship in case of connection between *Impairments data* and *User Interface characteristics* is similar to the *antecedent - consequent* relationship in the rules mechanisms we just described. Therefore, we can effectively utilize these to implement the connections.

5.6 Accessibility Framework

In order to provide accessibility in a systematic and generic way, the *Accessibility Framework* (see Fig. 5.2) is incorporated within our system which is based upon *Connecting Ontologies* of the contextual components and the related services for communication.

The interface to the *Accessibility Framework* is via *Accessibility Module* and the *Info-Viz Bridge Module* operating above the *Connecting Ontologies* which are formed as a result of rule execution across different contextual ontologies mentioned earlier (see workflow in Fig. 5.3). The rules to connect the entities from one ontology to another are stored in a text file. After the execution of rules new triples are generated which are stored separately for reuse. The user's request is forwarded to the *Accessibility Module* which then asks *Info-Viz Bridge Module* to suggest, for example, appropriate interface elements or visualizations. The interface is later adapted and made accessible

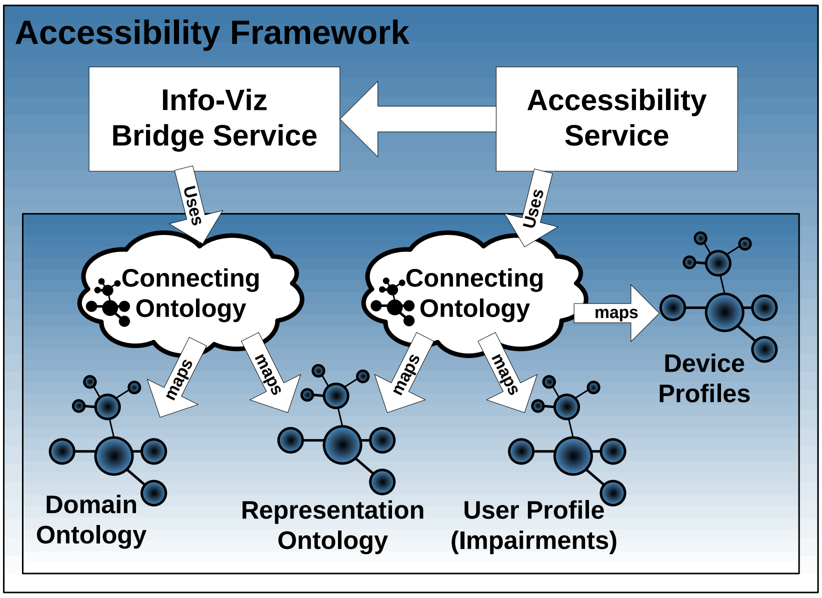

Figure 5.2: Overview of Accessibility Framework

using another connecting ontology which connects user impairments with device profiles and visualization recommendations through ontology design patterns and rules. Our accessibility service may also be useful for providing accessibility extensions to existing ontology based user interface frameworks such as Haystack (Karger et al., 2003).

5.6.1 Info-Viz Bridge Module

The recommendations of *Info-Viz Bridge Module* are based on the rules and patterns established between visualization / representation and tasks ontologies. It is important to note here that since the tasks are related with the application domain. Therefore those are modeled within the domain ontology. The recommendations are sent back to the *Accessibility Module*.

5.6.2 Accessibility Module

This module provides an interface to the *Connecting Ontology* between the *Impairments Data* and the *User Interface Characteristics*. It can be used in following two ways:

- ***CSS* adaptation for browsers:** In the ontology, we get the suitable *UI* suggestions against each type of *Impairment*. The *CSS* in use is

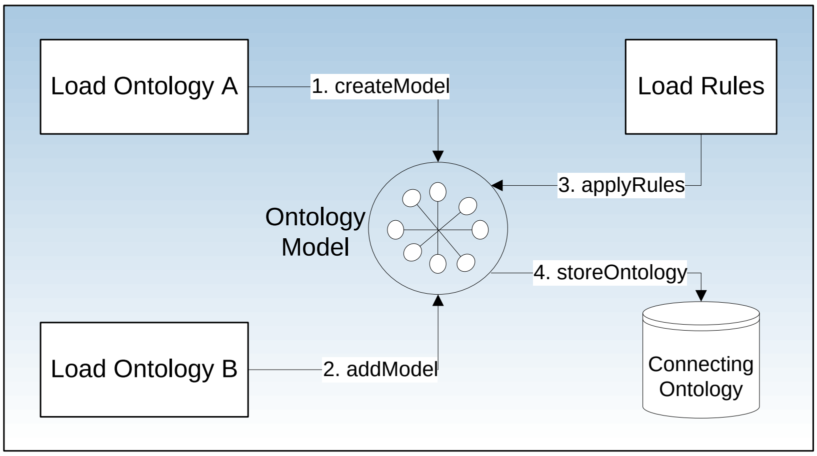

Figure 5.3: Workflow for forming Connecting Ontology

accessed using the *CSSOM Stylesheet Parser* (van Kesteren ed., 2007), and for each attribute appropriate value is mapped from the attribute values fetched from the *Connecting Ontology*. An alternate option is to build an ontology of stylesheet elements and attributes for connecting it directly with the *Impairments Ontology*. However, at present we do this mapping programmatically. In any case, *the related* CSS *can be updated on the fly*.

- **Access methods for use with visualization toolkits:** A number of methods are provided to retrieve the specific attributes and their values from the *Connecting Ontology* against an *Impairment Resource*. Alternatively, we can also retrieve all the *Impairment Resources* related with a specific *UI Resource*. The results can be used within the *API* of the visualization toolkit being used, as an input for rendering the visualization.

5.7 Summary

The need to connect heterogeneous knowledge domains is highlighted, and a novel way of their semi-automatic interconnection is suggested, called the *Connecting Ontology*. The representation of cause-effect relationships between two heterogeneous domains in a single ontology, become complicated and out of control with time and added information semantics. This type of combined ontology represents the tacit knowledge of domain experts. Inspite of its usefulness in a combined form, there ought to be some ways to explore the semantics of combining those domains in form of domain rules. The rules can then be implemented using the *Semantic Web Technology* in such a way so that the two domains are represented in separate ontologies and are only combined on demand by executing the rules. To get advantage of the outcomes for the end user, an *Accessibility Framework* is proposed and implemented which operates under our prototype *SemanticLIFE*.

Chapter 6

Connecting Impairments and User Interface Ontologies

The framework implementation task consists of developing ontologies of contextual components, and the related services for interacting with those ontologies in context of our system *SemanticLIFE*. The implemented system architecture is flexible enough to realize the different tasks as plug-ins using *Java* as the main development tool. Automation of this process consists of two major steps:

- Formal description of semantics for each component
- Formal description of semantics of consequences and effects of potentially interacting components on each other

As mentioned in Section 2.2 and Section 2.3, there are various *contextual components* which must be connected together for providing an *accessible* system. We follow a modular approach and try to connect two of the *contextual components*, i.e., the *user impairments data* and the *user interface Characteristics*. The connection between these is to be made in accordance with the *WCAG* to provide accessible user interface (Fig. 6.1).

Ontologies for the formal semantic description of user impairments, user interface characteristics, tasks related to a particular domain, and their interconnections are developed using recently emerging semantic web technologies. Ontologies can be developed and arranged by following different approaches such as taxonomic and faceted. For this research we restricted ourselves to a hierarchical approach which would lead to further approaches in future. According to our requirements we have followed suitable guidelines from various ontology development approaches mentioned in (Uschold and

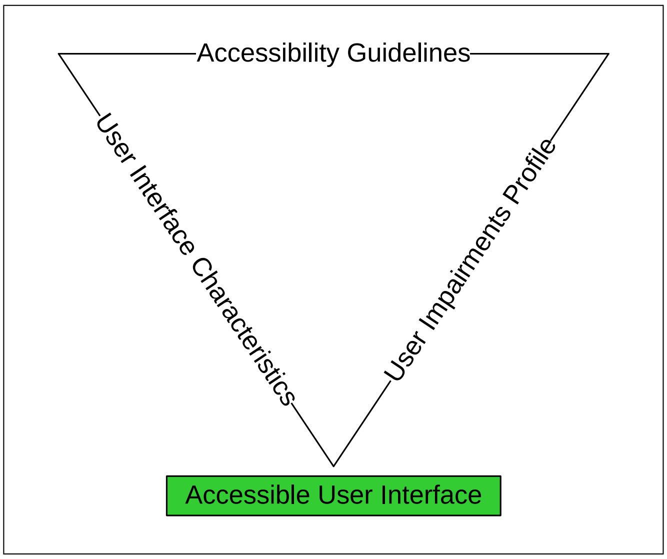

Figure 6.1: Towards an Accessible UI - The Essential Components

Grüninger, 1996) (Uschold and Grüninger, 1996) (Noy and Hafner, 1997) and (Kalyanpur et al., 2004).

The activity is divided in two steps, i.e., finalizing general ontology characteristics, and designing its contents. However the ontology development, especially the content design phase is to be iterative in nature. The method or the template we adapted for developing ontologies for *"Impairments"*, *"User Interface Characteristics"*, and *"Impairments - User Interface Characteristics"*, is mentioned in Appendix B).

6.1 Impairments Ontology

User impairments can be modeled in an ontology using the *Semantic Web Technology* which may be beneficial in many ways for the users of computer systems over the internet. We denote this ontology by the acronym ***ImpOnt***. At the beginning of this activity, important terms of the domain must first be collected and then their inter-relationships must be determined. In the domain of discourse the relationships can be defined permanently for some terms, and at times it is practical to infer the relationships based upon certain pre-conditions. The relationships or concepts may also change over time. After that, one can encode this conceptual schema of interconnected concepts using some ontology language like *OWL* (Bechhofer et al., 2004), and its instances can be populated according to the context in which it is being used. The medical terms can be found in abundance in libraries and internet sources like *Galen* (Rector et al., 2003), *SNOMED-RT*, *MeSH* and *UMLS* (UMLS, 1993). We also interviewed some physicians for this purpose [1]. Important terms with their brief definitions were written down. The intention was to find the concept hierarchies and their interconnections in the domain of discourse. A sample of impairments hierarchy based on existing medical categorization is given in Appendix C. Once this taxonomy is meaningfully and formally described in the ontology, then it is sharable with other users as well as with software agents for the collective benefit of all types of users and not only for users with special needs.

The *ImpOnt* ontology would contain the semantics about the impairments. The task consists of the following:

1. Development of *ImpOnt* ontology
2. *ImpOnt* plug-in service for querying this ontology

6.1.1 General Characteristics

It is important to know the purpose, and scope of the ontology. Then its possible interaction with other ontologies is specified, followed by a description of users, some motivating scenarios, and a list of competency questions.

Purpose and Scope

The ontology is related with physical and cognitive disabilities of the user. It contains the taxonomic structure of the impairments' concepts and their inter-relationships. Also, it is possible to grade the severity of each impairment on a predefined discrete scale. Ideally, the scale should be in accordance with standard medical practice and should be acceptable in the community of medical practitioners so that the ontology and the user's personal information management system could be more tightly integrated into everyday life other than the life of working on computer. The ontology can also be used for *eLearning* purposes. A few representative impairment concepts are described in the ontology which are sufficient to prove the concept. Later, it can be enhanced to include other impairments as well.

Interaction with other Ontologies

The interaction with the *User Interface Ontology* is planned which would be used for adapting the interface. Moreover, in our current prototype the impairments ontology will be required to interact with existing *UI* ontologies.

[1] Private communications with Zubair Kareem, M.D. Phy-Neurologist (Holyoke, MA)

Other possible interactions could be with *DOID* , *UMLS* , *ICD9CM* , and *MeSH* . Sample extracts from *DOID* and *MeSH* are:

```
[Term] id: Doid: 3203, name: Blindness of both eyes, impairment level not
  further specified, rank: 7, xref_analog: Umls_Cui:  C0271217, xref_analog:
  Icd9cm_2005: 369.00, xref_analog: Umls_Icd9cm_2005_Aui: A0241434, is_a:
  Doid: 3204

[Term] id: Mesh: A.01.456.505.420, name: eye, synonym: "eye" [], synonym:
  "ophthalmologic_effect" [], synonym: "ophthalmological_effect" [], is_a:
  Mesh: A.01.456.505 ! face, is_a: Mesh:A.09 ! sense_organ
```

It is to mention that we have already borrowed the terminological and taxonomic concepts from these resources. Our goal is to represent the semantics in *OWL DL* which none of these resources exactly fulfil according to our requirements.

Competency Questions

The implementation of *ImpOnt* ontology and the *ImpOnt* service should be made in such a way so as to answer the queries of various types such as:

- What is / are the related body parts given the name of the impairment
- What is the impaired side (right, left,...) given the name of the impairment
- What is the severity of the impairment (on a predefined scale)
- What are the perception cues which are affected by a given impairment name, and up to what degree (on a predefined scale)

6.1.2 Ontology Contents

ImpOnt ontology primarily contains the taxonomy of user impairments. The related body parts for all the impairments are described. Moreover, their effect on the information perception cues, as well as the capabilities is also described. Examples of impairments are color blindness, dementia, hemianopsia. Examples of body parts are eye, ear, brain. Examples of perception cues are attention, size, group. Examples of capabilities are Vision, Hearing, Touch. Normally, impairment severity is inversely proportional to the perception and the capability.

Primitive and Derived Concepts

A schematic overview of the ontology thus made is shown in Figure 6.2. It shows some of the primitive concepts. An excerpt from the related *OWL* file is shown in Listing 6.1. It is possible to execute the queries for retrieving the derived concepts such as to find all the *"left sided impairments"*, *"right sided impairments"*, *"any sided impairments"* and *"both sided impairments"*.

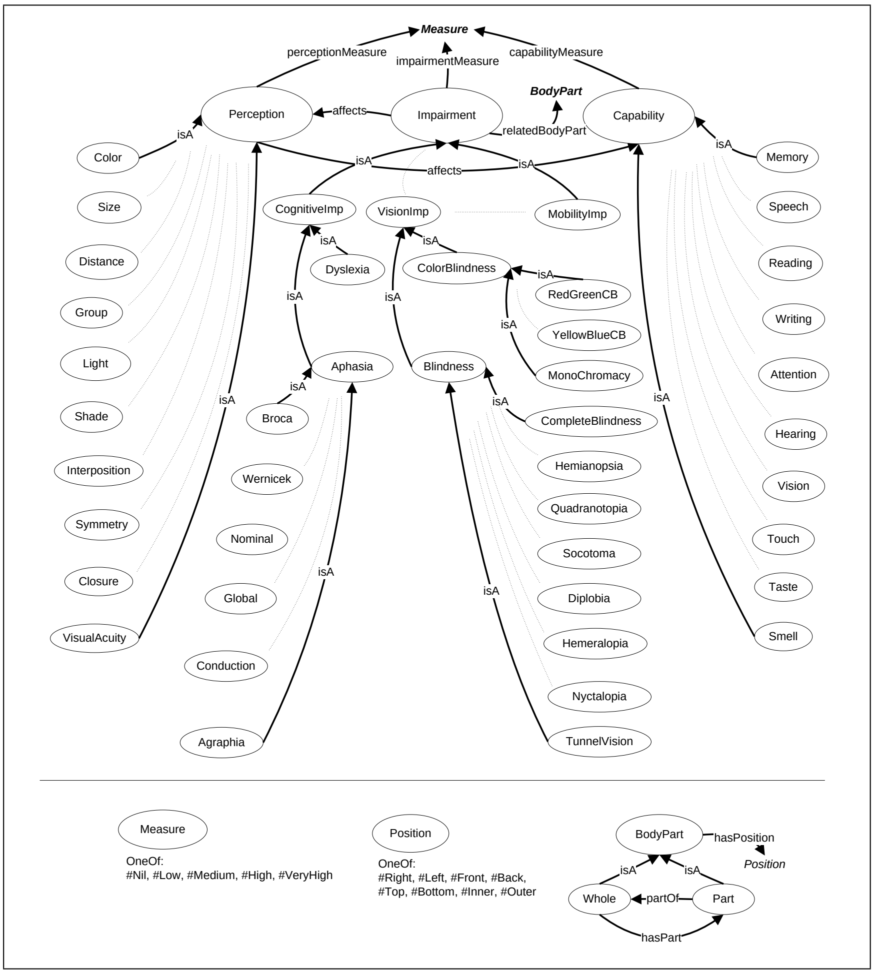

Figure 6.2: Schematic Overview of User Impairments Ontology

Listing 6.1: OWL DL for the Visual Acuity Concept in ImpOnt

```
<owl:Class rdf:ID="VisualAcuity">
  <rdfs:subClassOf rdf:resource="#InformationPerception"/>
</owl:Class>
 .....
```

```
 ....
   <owl:ObjectProperty rdf:about="#relatedBodyPart">
   <rdfs:range rdf:resource="#BodyPart"/>
   <rdfs:domain rdf:resource="#Impairment"/>
 </owl:ObjectProperty>
 <owl:ObjectProperty rdf:ID="capabilityAffectedBy">
   <rdfs:domain rdf:resource="#Capability"/>
   <owl:inverseOf>
     <owl:ObjectProperty rdf:ID="affectsCapability"/>
   </owl:inverseOf>
   <rdfs:range rdf:resource="#InformationPerception"/>
 </owl:ObjectProperty>
 <owl:ObjectProperty rdf:about="#affectsPerception">
   <owl:inverseOf>
     <owl:ObjectProperty rdf:ID="perceptionAffectedBy"/>
   </owl:inverseOf>
   <rdfs:domain rdf:resource="#Impairment"/>
   <rdfs:range rdf:resource="#InformationPerception"/>
 </owl:ObjectProperty>
 .....
 ....
 <Eye rdf:ID="Eye_Left">
  <hasPart rdf:resource="#Retina"/>
  <partOf rdf:resource="#VisionSystem"/>
  <relatedImpairment>
  <BinasalHemianopsia rdf:ID="BinasalHemianopsia">
       <relatedBodyPart rdf:resource="#Eye_Left"/>
       <relatedBodyPart rdf:resource="#Eye_Righ"/>
       <impairmentMeasure><Measure rdf:ID="Low"/>
   </impairmentMeasure>
        <affectsPerception>
     <VisualAcuity rdf:ID="VisualAcuity_Low">
          <perceptionAffectedBy
         rdf:resource="#BinasalHemianopsia"/>
          <affectsCapability>
           <Vision rdf:ID="Vision_Low">
            <capabilityAffectedBy
           rdf:resource="#VisualAcuity_Low"/>
            <capabilityMeasure rdf:resource="#Low"/>
           </Vision>
          </affectsCapability>
          <affectsCapability>
           <Capability rdf:ID="Capability_Vision">
            <capabilityAffectedBy
           rdf:resource="#VisualAcuity_Low"/>
            <capabilityMeasure rdf:resource="#Low"/>
           </Capability>
          </affectsCapability>
         <perceptionMeasure rdf:resource="#Low"/>
```

```
          </VisualAcuity>
         </affectsPerception>
        </BinasalHemianopsia>
        </relatedImpairment>
  <hasPartPosition rdf:resource="#Left"/>
 </Eye>
 .....
 ....
```

Concepts Restrictions

- *LeftSidedImapirment* $\subseteq$ *Impairment*
  LeftSidedImpairment $\equiv \exists$ *relatedBodyPart*($\ni$ *hasPosition*.{Left})

- *RightSidedImapirment* $\subseteq$ *Impairment*
  RightSidedImpairment $\equiv \exists$ *relatedBodyPart*($\ni$ *hasPosition*.{Right})

- *AnySidedImapirment* $\subseteq$ *Impairment*
  AnySidedImpairment $\equiv \exists$ *relatedBodyPart*($\ni$ *hasPosition*.{Left} $\cup$ *hasPosition*.{Right})

  We may also get *"AnySidedImpairment"* as a union of already derived classes such as:

- *AnySidedImapirment* $\equiv$ *LeftSidedImapirment* $\cup$ *RightSidedImpairment*

6.2 User Interface Ontology

Similar to the *User's Impairments Data*, the *User Interface Characteristics* can also be modeled in an ontology, denoted by the acronym ***UiOnt***. Optimal *UI* adaptation according to contextual components is necessary to increase the usability of the overall system. The *UI usability* is still largely subjective and is mostly based upon manual feedback from the users and *UI* experts. A lot of studies and theories related with user profiling and usability have been suggested but the formal connection with user interfaces is not yet fully achieved, partly due to missing knowledge representation formalism on both sides. *Semantic Web Technology* can be used to fill this gap which allows representing the information semantics in decidable *Description Logic* formalism thus making the knowledge base ready for information sharing and reasoning. Following these lines the ontology of user interface components is proposed which can be populated based upon the interface components while browsing the contents such as the web pages. Subsequently some representative usability measures can be entered manually which can later be automated based upon navigation on the interface vis-á-vis the specific task being executed.

The need to distinguish between objective and subjective measures of usability is emphasized in the comprehensive study of usability measurement practices in (Hornbík, 2006). The objective measures can be measured, discussed and validated, whereas the subjective measures are more about user's perceptions or impressions about the system in overall context of use. The context is a broad term which encapsulates many components such as the users, the tasks, the equipments and the environment as described in (Bevan and Macleod, 1994), while also highlighting the context of usability measurements such as effectiveness, efficiency and satisfaction. Effectiveness indicates if the intended goals of the overall use of the system are achieved; Efficiency shows the efficient utilization of resources such as time, money, persons and material; and Satisfaction signifies the user's acceptability of the overall system. Normally, it is not practical or nearly impossible to automate it fully. In specific application settings we may however prioritize the automation of the contextual components. Some important elements of the context are application specific information items, interaction devices, and user interface characteristics. Here, we concern ourselves with user interface and its relationship with usability. The *UiOnt* ontology would contain the semantics about the user interface characteristics. The task consists of the following:

1. Manual development of *UiOnt* ontology
2. *UiOnt* plug-in service for querying this ontology

6.2.1 General Characteristics

Purpose and Scope

The objective for defining this ontology was to semantically describe the user interface components in context of usability. Information about the model *UI* components is taken from SWT Toolkit (Northover and Wilson, 2004) and the model *UI* usability measures from the literature (Hornbík, 2006)(Kieras, 1988)(Norman, 2000)(Holzinger, 2005) which emphasize among other things the tasks, culture and capabilities of the users. The *UiOnt* ontology would contain the taxonomy of *UI* components. Each component will have standard usability gradings according to a user without any special needs in normal circumstances. For example, for a text field these might correspond to font size, color, style and manoeuvrability on the interface. Atomic components may also be combined to form composite components with their modified usability grading for the composite component.

Interaction with other Ontologies

The interaction with *ImpOnt* ontology is planned and would be useful for adapting the interface. Moreover, the concepts can be aligned with existing taxonomy of *UI* components. From the literature we did not find any significant semantic structure of these components. That is also one of the reasons, to develop this ontology by ourselves for test purposes. The input is based upon the user interface components as described in *SWT Toolkit* (Northover and Wilson, 2004).

Competency Questions

The implementation of *UiOnt* ontology and the *UiOnt* service should be made in such a way so as to answer the queries of various types such as:

- Find the part-whole relationship of user interface components
- Find the attributes of a component and their values (according to predefined usability scale for a normal user in normal conditions)
- Given specific attribute name(s), find the related UI component(s)

6.2.2 Ontology Contents

It contains the taxonomic structure of user interface components and their characteristics. The relationship of components with each other is also described. The ontology is expressive enough to represent the containment hierarchy of *UI* components, and other useful semantic information. Also, the relative position within a containment hierarchy can be stored. For example, semantics to see if a component is only an output widget, a selection widget or a navigation widget. The selection is further categorized into boolean selection (such as radio buttons, check boxes) and multiple selection (such as combo box, spinner, menu bar). Every *UI* component is valued with some quantitative usability measure. The *structural comprehension* of some components such as radio button and check box is more readily understood than a combo box or a spinner. Similarly, the judgment about the *functional behavior* of a component would vary from user to user. The *feedback* returned by interacting with a checkbox may feel much better than the feedback from a toggle button. Similarly, the ease of *user control* over a component varies.

Primitive and Derived Concepts

A schematic overview of the ontology thus made is shown in Figure 6.3. Some of the primitive concepts are shown. An excerpt from the related *OWL* file is shown in Listing 6.2. It is possible to formulate and execute queries for retrieving the derived concepts such as *"good usability components"*, *"fair usability components"* and *"fair user control components"*.

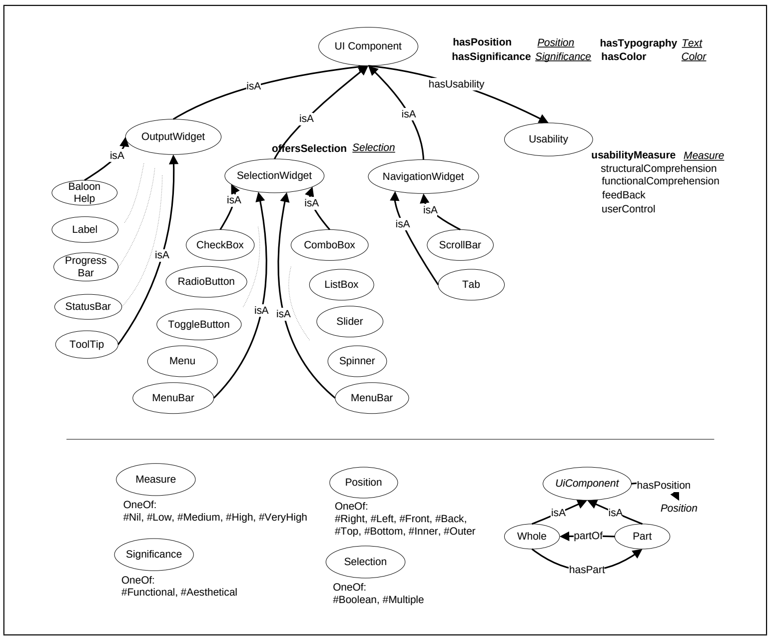

Figure 6.3: Schematic Overview of User Interface Ontology

Listing 6.2: OWL DL for a UI Component in UiOnt

```
<owl:Class rdf:about="#Text">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#UiComponent"/>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty>
        <owl:TransitiveProperty rdf:ID="partOf"/>
      </owl:onProperty>
      <owl:someValuesFrom>
        <owl:Class rdf:ID="ComboBox"/>
      </owl:someValuesFrom>
    </owl:Restriction>
  </rdfs:subClassOf>
```

```
</owl:Class>

<owl:ObjectProperty rdf:ID="hasLegibility">
  <rdfs:domain rdf:resource="#Text"/>
  <rdfs:range rdf:resource="#Measure"/>
  <rdfs:comment>equivalent concept of structuralComprehension.
  </rdfs:comment>
</owl:ObjectProperty>

<owl:Class rdf:ID="UiComponentUsabilityGood">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#UiComponent"/>
  </rdfs:subClassOf>
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty>
        <owl:ObjectProperty rdf:about="#hasUsability"/>
      </owl:onProperty>
      <owl:hasValue>
        <Measure rdf:ID="Good"/>
      </owl:hasValue>
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>

<owl:Class rdf:ID="TextStyle">
  <rdfs:subClassOf>
    <owl:Class rdf:ID="Text"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="TextFont">
  <rdfs:subClassOf>
    <owl:Class rdf:about="#Text"/>
  </rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID="TextSize">
  <rdfs:subClassOf rdf:resource="#Text"/>
</owl:Class>
 .....
 ....
<TextSize rdf:ID="TextSize_18">
  <fontsize rdf:datatype="http://.../XMLSchema#int">
  8</fontsize>
  <hasLegibility rdf:resource="#Good"/>
</TextSize>
<TextSize rdf:ID="TextSize_10">
  <hasLegibility rdf:resource="#Good"/>
  <fontsize rdf:datatype="http://.../XMLSchema#int">
  10</fontsize>
```

```
</TextSize>
<TextStyle rdf:ID="TextStyle_Bold">
  <hasLegibility rdf:resource="#Good"/>
</TextStyle>
<TextStyle rdf:ID="TextStyle_BoldItalic">
  <hasLegibility rdf:resource="#Fair"/>
</TextStyle>
 .....
 ....
```

Concepts Restrictions

- *GoodUsabilityComponent* $\subseteq$ *UiComponent*
  GoodUsabilityComponent $\equiv$ *UiComponent* $\cap$ ($\ni$ *hasUsability*.{Good})
- *FairUsabilityComponent* $\subseteq$ *UiComponent*
  FairUsabilityComponent $\equiv$ *UiComponent* $\cap$ ($\ni$ *hasUsability*.{Fair})
- *FairUserControlComponent* $\subseteq$ *UiComponent*
  FairUserControlComponent $\equiv$ *UiComponent* $\cap$ ($\ni$ *userControl*.{Fair})

6.3 Impairments - User Interface *Connecting Ontology*

It is denoted by the acronym ***ImpUiOnt*** and it would contain the semantics about the impairments and the corresponding suitable user interface characteristics. The task consists of the following:

1. Development of *ImpUiOnt* ontology
2. *ImpUiOnt* plug-in service for querying this ontology

The basic concept of Connecting Ontology between Impairments and UI was explained in (Karim and Tjoa, 2006b).

6.3.1 General Characteristics

Purpose and Scope

The ontology is about user's physical and cognitive impairments, and the related consequences on the *UI*. It will be used in context of user's personal information management system. The objective is to exploit the impairments data so that user's interaction with the system is optimized in terms of improved *UI*. The aim of this ontology is to make improvements for all the disabilities or impairments of the users in a generic way instead of focusing only on the stereotypical disabilities. By exploiting this generic ontology, improved personalized interaction for all could be possible to accommodate diversity. It is a domain specific ontology and will not cover general world or common sense knowledge in the initial attempt.

Interaction with other Ontologies

This ontology provides the user interface recommendations according to the user's impairments profile. Therefore the obvious candidates for interaction are the style sheets for use in browsers, and also the ontologies of visualization toolkits, if they exist already. Other than that, the ontology can also be very useful for integrating with some applications related with the rehabilitation efforts for people with disabilities.

Users and Usage Scenarios

The ontology will be used by the end user or his / her caregiver in case of severe impairments like cognitive problems. User interaction is not only limited to traditional UI part. This ontology is usable in many ways, such as the following:

a. The ontology will be used for customizing / adapting the UI according to user impairments. Each interface component has a certain affordance of use (Norman, 2000). If the user is unable to use it in that way, then it should not be part of the UI. Instead a component with better affordance would be suggested for this user, e.g.,

   - Color blindness can be Red-Green, Blue-Yellow, and Monochromacy (complete inability to distinguish any color). The confusing colors on an interface can be avoided for a particular type of user.
   - Visual acuity refers to the clarity of one's vision, a measure of how well a person sees. Font size can be adjusted according to the user's visual acuity.
   - Blindness in one half of the visual field. For this user, the information should only be presented on the better half of the screen.

b. In tourism domain, the ontology can provide accessibility for a better travel planning. If user has some mobility impairments, then while searching a route from point A to point B the availability of accessible lifts and restrooms can be shown on transits, and also the corresponding time calculations for the journey (e.g., accessibility of http://www.vor.at/).

Competency Questions

A number of queries are framed which must be answered by our ontology. The implementation of *ImpUiOnt* ontology and the related service should be made in such a way so as to answer the queries of various types such as:

- Avoidance of confusing colors for particular type of users color blindness
- Font adjustments according to users visual acuity
- Information presentation on the better part of the screen for a user suffering from Hemianopsia (absence of vision in half of visual field)
- Given an impairment name, find the suitable UI components with attributes and their values
- Given an UI component name with associated attributes, find the impairment name(s) where it can / can't be used with

  Or more specifically speaking;
- Is UI component (such as vertical scrollbar) suitable for the user? Otherwise what is the alternate interface component?
- What is suitable font size?
- Which colored control buttons are suitable?
- What is the most suitable screen area for presenting information (right, left or central)?
- Is lift facility available on *Karlsplatz* underground station?
- Are textual descriptions available using Braille for *Albertina Gallery*?

Finally, as part of evaluation, all the queries should be satisfied and should give correct, consistent, and reliable results without any regression.

6.3.2 Ontology Contents

The contents of this ontology will be derived from the contents of previously described *ImpOnt* and the *UiOnt* ontologies.

Important Terms and their Organization

The terms for this ontology are the properties *"suggests"* or *"suitable"* and the contrary, i.e. *"notsuggests"* or *"notsuitable"*. No significant organization was required. The actual semantics of the organization between the *UI Components* and the *Impairments Data* is made via rules, which are described later. However, the knowledge for these rules is based upon the taxonomy of a representative set of impairments given in Appendix C.

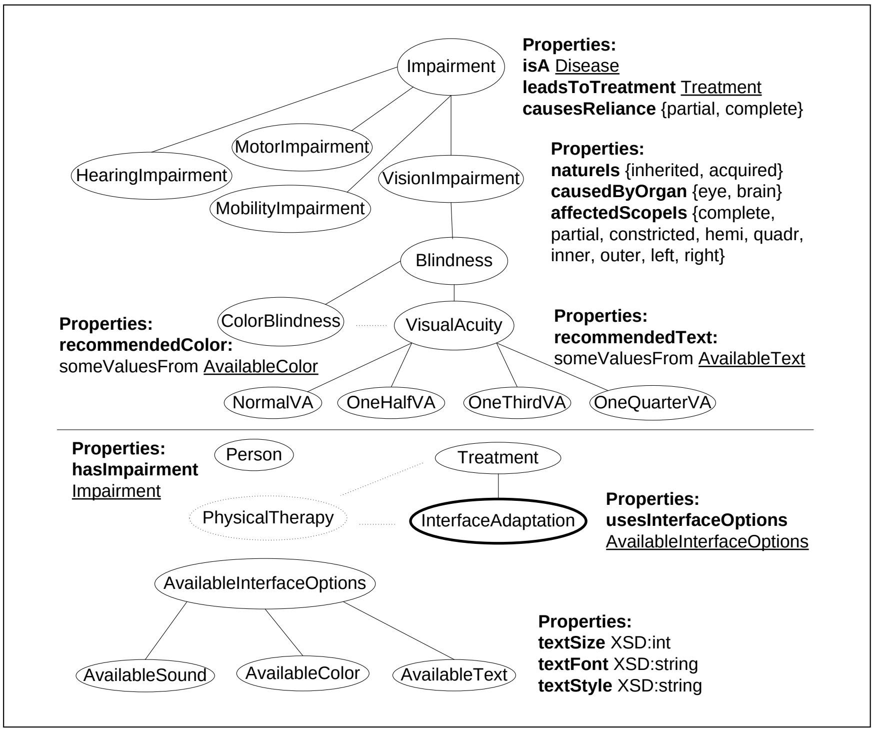

Figure 6.4: Part of Impairment-User Interface Ontology

Classes and Class Hierarchy

Extending the work from (Hadzic and Chang, 2005) (see Fig. 2.2) an ontology is envisaged as shown in Fig. 6.4. This is to be implemented in *OWL DL* due to available *DL Reasoners*. The main extension is the *InterfaceAdaptation* (see Fig. 6.4) as another treatment for users with disabilities. “Impairment” is a type of disease which needs to be addressed. As a result of impairment the concerned person can become partially or completely reliant on some one. The sub-classes of “Impairment” and other classes are managed with the help of given properties. As a sample case, visual acuity is taken care of by binding it to appropriate text sizes. The sub-classes of "VisualAcuity" are associated with "AvailableText" by specifying restriction for its property "recommendedText". While entering the user's impairments data (using instance of "Person") user's visual acuity instances are linked with suitable text sizes. Then it is used for displaying text on the interface for this user.

This is achieved by making manual assertions in *OWL DL* between the concepts from *ImpOnt* and *UiOnt* ontologies. Below, we show that by using rules, how it can be automatically achieved.

Connection **Rules**

The connections are made by rules via *Jena Inference API* (Reynolds, 2007). For example:

- *Low perception implies suggesting high usability components*
  (?x rdf:type imp:VisualAcuity) (?x imp:perceptionMeasure imp:Low)
  (?y rdf:type ui:UiComponent) (?y ui:hasLegibility ui:Good) →
  (?x eg:suggests ?y)

- *High perception implies suggesting fair usability components*
  (?x rdf:type imp:VisualAcuity) (?x imp:perceptionMeasure imp:High)
  (?y rdf:type ui:UiComponent) (?y ui:hasLegibility ui:Fair) →
  (?x eg:suggests ?y)

- *High rheumatism implies suggesting easily operable components*
  (?x rdf:type imp:Rheumatism) (?x imp:impairmentMeasure imp:High)
  (?y rdf:type ui:UiComponent) (?y ui:userControl ui:Good) →
  (?x eg:suggests ?y)

6.3.3 Results

Upon execution of rules the suggestions are generated (Fig. 6.5) which can be stored for reuse until there are some further changes in the participating ontologies.

6.3.4 Results Exploitation

Interface Adaptation

These suggestions are processed for adapting the *CSS* (see Fig. 6.6, 6.7,6.8. For example, there is a range of recommended text sizes for each type of visual acuity. The discrete values in each range are mapped to text size attributes for header tag *"h"* in a *CSS*. The highest suggested *TextSize* is

```
<rdf:RDF
    xmlns:co="http://www.ifs.tuwien.ac.at/ontologies/co#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdf:Description rdf:about="http://www.ifs.tuwien.ac.at/ontologies/imp#VisualAcuity_High">
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_09"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_26"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextStyle_Italic"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextStyle_BoldItalic"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_08"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_24"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.ifs.tuwien.ac.at/ontologies/imp#Rheumatism_High">
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#LabelledButton"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#ComboBox"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.ifs.tuwien.ac.at/ontologies/imp#VisualAcuity_Low">
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_18"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_10"/>
    <co:suggests
rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextFont_TimesNewRoman"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextStyle_Bold"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_22"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_11"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#TextSize_20"/>
  </rdf:Description>
<rdf:Description rdf:about="http://www.ifs.tuwien.ac.at/ontologies/imp#ColorBlindness_RG">
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Grey"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Orange"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Cyan"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Blue"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Yellow"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.ifs.tuwien.ac.at/ontologies/imp#ColorBlindness_YB">
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Grey"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Orange"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Cyan"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Green"/>
    <co:suggests rdf:resource="http://www.ifs.tuwien.ac.at/ontologies/ui#Color_Red"/>
  </rdf:Description>
</rdf:RDF>
```

Figure 6.5: Suggestions for Specific Impairments Generated by Applying Rules

mapped to *"h1"*, the next *TextSize* to *"h2"* and so on. For the time being, this mapping is done programmatically using *CSSOM Style Sheet Parser* (van Kesteren ed., 2007).

6.3.5 Consequences

The consequences of this ontology are numerous. Primarily, it will be helpful in adapting the UI for a specific user, e.g., specifying a suitable text size. It can also be used in deducing the best match for a user with multiple impairments. For example, if a user has a very low visual acuity and also can not focus on the left part of the screen then it is a composition of impairment classes and related interface options. Also, the historical data generated by the users would help to get an insight about the evolving cause-effect relationship between the impairments and the computer interfaces.

A next step to be performed is the connection between *Impairments* ontology and style sheet elements and attributes. The result set in the form of *RDF* triples is an ontology in itself which is sharable and process able by

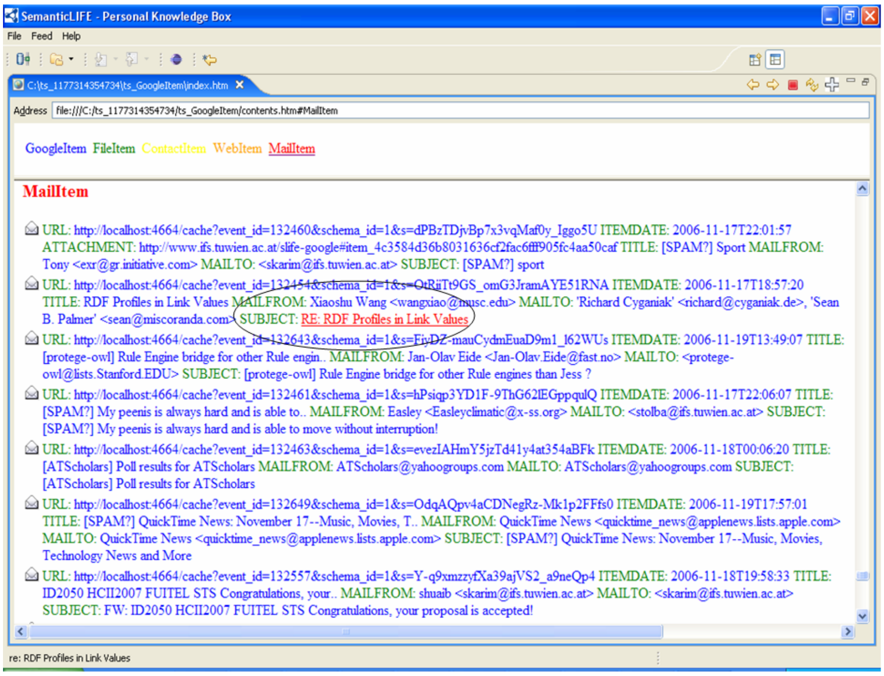

Figure 6.6: Confusion Due to Red-Green Color Blindness

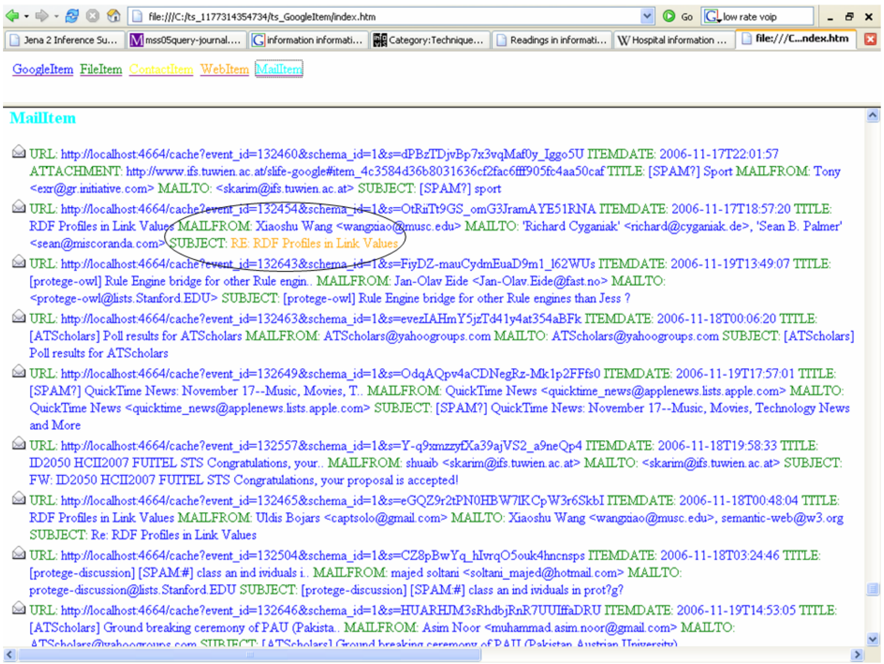

Figure 6.7: Solution for Red-Green Color Blindness

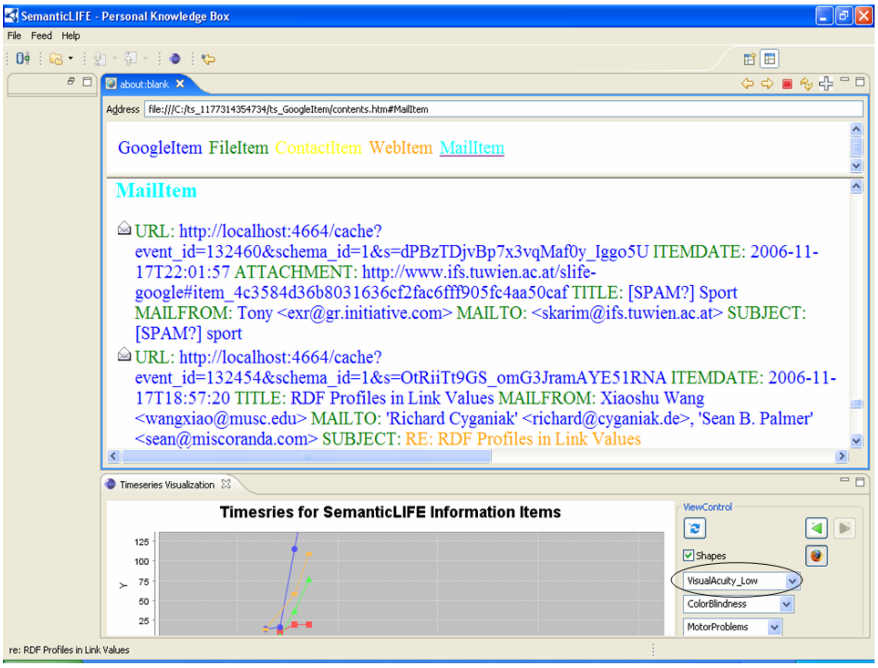

Figure 6.8: Solution for Low Visual Acuity

Semantic Web tools for any useful purpose.

Also, the *ImpOnt* and *UiOnt* ontologies, and the method of *Connecting Ontologies* could be usefully exploited towards the automation in *Usability Engineering* (Holzinger, 2005) in general.

6.4 Summary

Bottom up approach of incorporating accessibility is a complex and unmanageable task which consumes a lot of resources. Consequently, it has proved to be a repulsive factor in itself for the software producers in providing universally accessible tools in general (Keates, 2006). The presented approach tackles the problem space components' interactions from a holistic point of view using *Connecting Ontologies* while preserving the freedom of components' reuse by having individual ontologies of their own. The Ontology design patterns are successfully employed in our system which makes the approach more convincing for the practitioners. The *PerceptionEffectPattern* from the collection of ontology design patterns defined in Chapter 4, is implemented within our *Accessibility Framework*. It is shown how *UI* can be successfully adapted based upon the user's impairments data. On similar lines, other patterns may also be implemented.

Chapter 7

Exploitation of *Connecting Ontologies* - Motivating Scenarios

Some motivating scenarios are described where our proposed approach can be used effectively. The final results are not provided because the implementation of these scenarios is still in progress.

7.1 A Person Equipped with *SenseCam*

The scenario was explained in (Karim and Tjoa, 2006a). This is about the working life of our blind colleagues in office environments. During business meetings, blind persons are not able to see the meaningful movements, and facial gestures of the participants. The formal meeting minutes and the participants' conversation during the meeting normally lack this important feedback in order to determine who is in favor and who is against their proposed suggestions. This is crucial in business negotiations, where one has to convince people and do lobbying for winning the business case in upcoming meetings. Today devices already exist for instantly and seamlessly capturing the snapshots everywhere. The proposition suggests data capture using a similar device called the *SenseCam* [1], and then making these snapshots accessible for all, especially for people with severe vision problems.

[1] http://research.microsoft.com/sendev/projects/sensecam/ (5th September 2007)

7.1.1 Role of Associations

The archiving of meeting proceedings is considered to be very useful for the immediate benefit of the participants. However, exploring the associations between the meeting room constituents (participants, and objects within the meeting room) vastly increases its benefits. The *multiple forums and vocabularies*, and *multimedia information integration* are marked as two of the technical research challenges by (NIST, 2006). The important clues for meeting recognition are *What is being discussed, Who is in action, To whom one is talking to, When the meeting is taking place, and Where* (Yang et al., 1999).
The above questions can be significantly answered by managing the associations of user's information items using ontologies. Our prototype system *SemanticLIFE* can explore and manage the following types of associations existing within user's lifetime information:

- Firstly, there exist structural associations. Each information item has an inherent association with its structural meta-data. For example, an email is associated with header fields such as Subject, From, To, Received and Sent Dates.
- Secondly, associations could be asserted using manual annotations. For example, a *contact information item "X"* can be manually associated with a *project proposal document "Y"* based upon the *collaboration done* by *"X"* on *"Y"*.
- Thirdly, using sophisticated techniques of textual and multimedia content analyses, further associations in the concerned ontology are possible. The examples are the possible association of an email message containing the word "ICCHP" in body with some web page having title "ICCHP 2006", association of information items existing in the same time slice (editing a project proposal document and browsing the web for related information in parallel), association of information items generated from the same location or related with same location (a picture taken and a telephone call made from the same place).

Also, significant benefit is achievable by finding the active and passive project participants based upon the presence or absence in related project meetings which can be calculated after the pictures are annotated for relative movements and gestures of the participants. The next step is to present these associations in an accessible way for people with special needs by applying appropriate accessibility guidelines.

The point of our interest is to apply the accessibility criteria also at the contents (making accessible the associations of captured snapshots) and not just at the presentation level for facilitating the exploration of associations for people with special needs.

7.1.2 Background

The importance of the domain is highlighted by the presence of several projects related with meeting room recognition technologies such as "The Meeting Recorder Project" by (ICSI, 2000), "The Meeting Recognition Project" by (NIST, 2006), "Meeting Browser" and "Computers in the Human Interaction Loop" by (ISL, 2001), "Augmented Multi-party Interaction" by (AMI, 2007), and "Interactive Multi-modal Information Management" by (IM2, 2007), just to name a few. Multimedia techniques are used in (Yang et al., 1999) to track the meeting with participant's ID using color appearance, face id, and speaker id. The corpus based framework in (Reidsma et al., 2005) describes how the meetings are modeled in layers, and how the annotation could be used for meeting recognition. The work done by (Howard D. Wactlar, 2003) is highly related with our work. They have used image processing techniques for monitoring and tracking the user activities. To achieve the same goal we are focusing on linking the users information items with each other based upon meta-data. This will also help in resolving the synchronization issue between different data sources, a problem mentioned by (Howard D. Wactlar, 2003). The dialog act labeling guide (Shriberg et al., 2004) describes the audio dialog structure and the annotation system used. Our approach is different since it is concerned with the usage of annotations instead of audio dialog for exploiting the gestures and participants movements. It is possible to identify and annotate the meeting room constituents by using ontologies of interconnected information items. We are optimistic that this component will supplement the capture and recognition of meeting room knowledge exchange significantly. Examples of available image annotation tools for the semantic web are Flickr2RDF (Flickr2RDF, 2007), PhotoStuff (Mindswap, 2003), M-OntoMat-Annotizer (acemeDia, 2004). Some of these convert the annotations into owl file like flick2rdf. The supported annotations are mostly user comments and are not sufficient for our purpose of capturing the meta-data. The recent use case document by W3C (van Ossenbruggen et al., 2007) describes the issues and challenges in carrying out manual, semi-automatic, and completely automatic annotation of images. Our approach is in line with their vision as we are starting with semi-automatic annotations and keeping our system architecture flexible to integrate the efforts done by image processing community in future. For applying our strategy the meeting snapshots must be captured in sufficient detail. *SenseCam* which is a badge size wearable camera is capable of doing this under user control. It can take pictures of the meeting room based upon any small change in environment (location of objects, light and temperature). It is in use by *Microsoft* in their research project MylifeBits (Gemmell et al., 2006) with promising results highlighting new challenges in managing personal information in various domains. *We intend to use it for blind by exploiting the annotation mechanism in our prototype SemanticLIFE.*

7.1.3 Sample Use Cases

Blind person sitting in a business meeting

During a meeting, there are many possible movements, postures and facial gestures that are made by the participants such as leaving or entering the room, sitting down, standing up, whispering with someone while leaning, relaxing on the chair, sitting alert, hand gestures by the participants, talking on telephone, working on laptop, and apparently sleeping. Automatic capture of these movements would help to anticipate the mood of the participants during discussion.

- The blind person wears the *SenseCam* during the meeting. So the pictures of the whole proceeding are taken automatically.
- Pictures are uploaded in SemanticLIFE repository as our file upload data feed.
- Retrieval of day's pictures from the repository and identification of participants either manually by the caregiver or automatically (Jeon et al., 2003) using multimedia analysis plug-ins with possible help from ontologies.
- Annotation of pictures by the caregiver based upon the gestures / movements of the constituents (participants and other objects in the vicinity). Initially, it is assumed that the caregiver is already informed about the identities of the constituting objects. Later, the identities should be matched against similarity using ontologies.
- Enrichment of associations; For example, the gestures by a participant would update his / her contact profile for a particular project meeting. This will give useful information to the blind user about this specific participant in future meetings.

The system is usable in domains other than the project meetings and also for a more diverse range of special needs (Pühretmair, 2004). The following scenario is useful for the blind persons as well as for users with mobility impairments.

Blind person visiting the city

In this case, the captured data is much broader. The pictures of our interest can also be those of stationary objects like restaurants, information counters, monuments, facilities coming on way to the destination.

- Blind person equipped with *SenseCam* is traveling from one station to another. The pictures taken would most probably include the facilities like restrooms, lifts, ramps about which the blind person is not aware of on his / her initial visit.
- Similar to the above scenario, the pictures would be annotated by his / her caregiver, and associations would be made using ontology.
- Before making the subsequent visit, the blind user can consult his / her route planning. Based upon the associations made in the previous step, the system could present to him the information about availability of accessible facilities en route and now the travel planning and travel itself could become more accessible.

Since, the SenseCam capture rate is about 1 picture/sec. Therefore, for an hour proceeding, there are 3600 pictures which is a big number. The annotation mechanism should be very user friendly so that the caregiver easily annotates all the pictures.

7.1.4 Proposed Approach

An ontological approach is adapted for representing the different axis for categorization and grouping of the information items.

Ontology for Meetings and Projects

We have assumed "Meeting" to be a specific type of activity or event within a project. Therefore, we have defined our Meeting Ontology in context of the Project Ontology. For projects, already existing project related ontologies such as (UMBC, 2007) and (DOAP, 2006) can be reused after modification. We propose an extended ontology (see Fig. 7.1) for specifying concepts related with projects and the binding with our feed items. An effort is made to keep it coherent with the *CODeP "Pattern for Roles and Tasks"* of our reference foundational ontology *DOLCE* as described in (Gangemi, 2005).

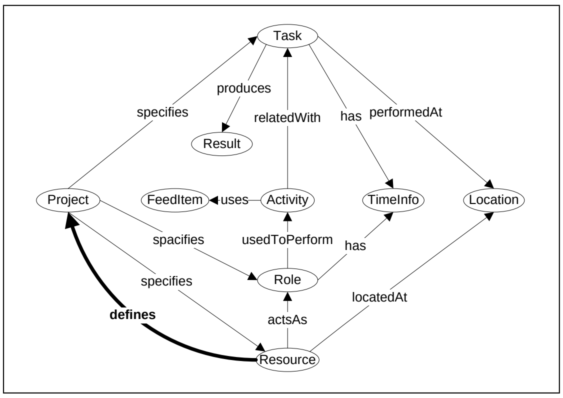

Figure 7.1: Projects Ontology Overview

The ontology depicts the required concepts to fulfill our scenario. The "Meeting" concepts and their relation with the "Projects Ontology", are described below:

```
Meeting                     AgendaItem            Posture
 meetingType XSD:string      priority XSD:int      postureName XSD:string
 lasts XSD:time              lasts XSD:time        adoptedBy HumanResource#
 location XSD:string         relatedTask Task#     lasts XSD:time
 convenedBy HumanResource#   proposedBy HumanResource#
 relatedWith Project#
```

There can be different "MeetingTypes" such as highly formal, moderately formal, informal, research brainstorming, regular departmental meeting and coordination meeting. Meetings are carried out in context of a specific project. Examples of "ProjectType" are software development, construction, tourism and entertainment. A "ProjectType" has specific "Roles" such as manager, analyst, developer, tester, and integrator depending upon the project responsibilities. The "Tasks" are associated with the "Roles" to carry out specific responsibilities, and are discussed in meetings under the specific "AgendaItems" lasting some "TimeFrame". The meeting agenda is typical for specific type of project and specific type of meeting. The meeting is convened by someone, and the agenda items are also proposed by someone. The meeting has a certain "Location". The person also has a location while attending the meeting (inside the meeting room or at a remote place), the meeting has a "TimeFrame", and so do the agenda items which are normally prioritized and are assigned certain "PrioritySequence". Each "Agenda Item" lasts for some duration, and the participants assume "Postures" lasting some timeframe throughout the meeting.

Categorization of Pictures

The big amount of pictures is categorized into distinct parts based upon meaningful criteria such as given in Fig. 7.2. The parts thus holding a smaller amount of photos are annotated, which is a relatively manageable task. The criteria to categorize the bulk of information items is devised in such a way so as not to loose any meaningful information about the whole meeting proceedings, yet being able to retrieve everything based upon a group of criteria, like a data cube (Gray et al., 1997).

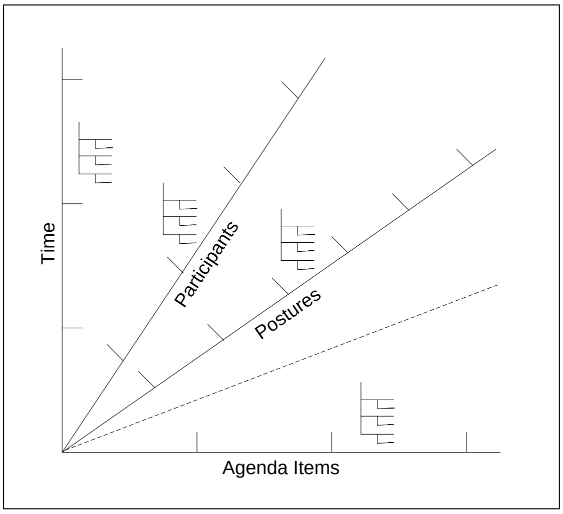

Figure 7.2: Information Distribution Over Multiple Axes

The primary criterion is time distribution. The secondary criterion is based upon the participants' tracking (their identification and relative postures). Another criterion is to associate the issues or agenda items discussed during the meeting with the information items. There can be other criteria as well, based upon the evolving semantics and the needs of the user.

Distribution according to time: A long meeting can be heuristically broken down into "n" minute's duration of "m" parts each. For this purpose the time stamp available in the *Exif* header of the pictures is used. The process is carried out automatically by the analysis plug-in in our system based upon predefined values in the configuration file.

Distribution according to participants' tracking: The tracking of participants is very important because the issues discussed or decided in a meeting can be meaningfully related with it. There can be two steps, namely the person's identification and the posture's identification. For example, distribution of pictures based upon names of the persons. Initially it is sufficient to tag the exit and entry of a participant which is possible by following a manual protocol, or exploiting some available identification technologies like *RFID*. Later, some of the major postures lasting for the duration of parts can be identified and annotated.

Distribution according to agenda items: The nature of the "AgendaItems" is specific from project to project. Some agenda items will be common in many projects. There is a small chance that the items are unique for a common project and meeting type.

Workflow for Managing Associations

The information items are stored as *RDF* triples in the repository against a base ontology. This ontology specifies the header fields for each information item as its properties. The associations related with different criteria as mentioned earlier, can be carried out under user control. For that purpose, the implementation in separate components is planned, as shown in Fig. 7.3.

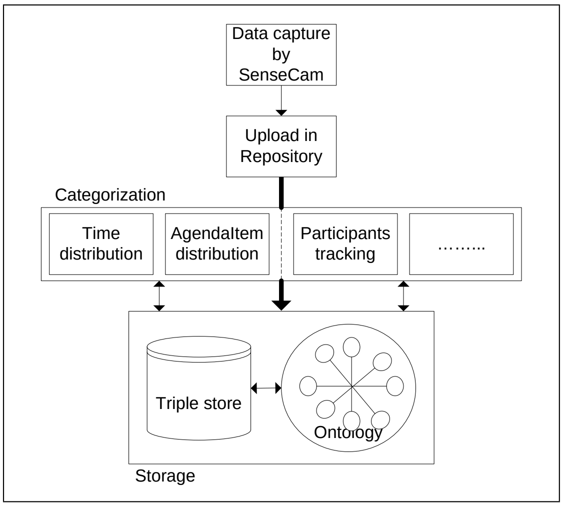

Figure 7.3: Workflow of Annotation Subsystem

The *UI* plays an important role for convenient interaction and annotation of pictures. The different types of information items from this triple store are displayed differently, say in different colors and on different graphs. According to user studies, the whole day is divided into 4 to 15 activities out of which the meetings are generally 2 to 5 for users under study. The activities with their times and durations may already be described in the calendar. Using the range sliders the user is able to select slices of time line during which a meeting has taken place. At times, the planned activities overlap with performed activities. Then the user can handle it by adjusting and confirming the slider positions on the *UI*. In the selected time slice, the user can filter out the items other than the pictures. Each selected time slice may correspond to "Subject" of the appointment in calendar. If not specified in calendar, then a right click of the mouse would enable the user to put the items of the selected time slice into a new named collection, say *SlifeMeeting* of type "Meeting" which may be connected to Meeting Ontology specified elsewhere on the web. Participants for this meeting are retrieved from calendar, if specified already. Otherwise, right clicking the *SlifeMeeting*, would enable the user to enter meeting participants.

Structural enhancement of information items: By visualizing the retrieved pictures, the components of the meeting room constituents are described by the caregiver. The picture *Exif* header specifies Camera-specific properties (such as its make, model, sensing method and lens size) and data about Image-specific properties (such as creation date, image resolution, height and width). The structure can be enriched to take into account the constituents by describing who is present in the picture and their postures. This is a laborious and time consuming task. By using intuitive *UI*, and ontology of interconnected information items, the task can be made more suggestive and convenient for the user.

Manual associations with other information items: It is possible that some other activity like a telephonic conversation, chat session or web browsing was carried out during the meeting. These activities may be related with the meeting under progress. Using manual annotations the individual pictures or collections of pictures can be associated with each other or any other information item in the repository or a concept. For example, the user can rate some named collections as highly or moderately useful, or useless. It will be beneficial for analyzing the time usage by the user and other meeting participants.

Dynamic associations: Once the ontology (see Fig. 7.1) instances are populated, then dynamic association of entities is possible. The information extraction techniques will be applied to the contents of fed information in cases where linking of fields in item header is insufficient to firmly establish the associations. For this purpose the concepts or the key terms are mapped with the ontologies. The links established through the ontology would make it possible to explore many things such as the historical behavior of a person in specific type of meetings and the analysis of specific agenda items with the duration of discussion.

7.1.5 Connecting Project Ontology and Data Feed Ontology

It is to note that the triples against the Project Meeting Ontology are populated separately from SemanticLIFE system. We only populate the Data Feed ontology. The main classes in Data Feed ontology are shown in Fig. 7.4.

Information Item Information item on user's computer
Attributes {Description, Date, URI, Size}
- **Email** Emails sent or received by the user
  Attributes {Subject, Sender, Receiver, Attachment(s), ...}
- **Chat** Chat sessions carried out by the user
  Attributes {Participants, Start DateTime, End DateTime,...}
- **File** Any kind of file on user's computer
  Attributes {Name, Path, URL, ...}
- **Web Page** Pages browsed by the user
  Attributes {Title, URL, ..}
- **Contact** Items in user's address book
  Attributes {Name, Address, Tel, email, Contact Type(family, business, friends,..)...}
- **Calendar** Items from user's calendar
  Attributes {Title, Date, Location, Audience,...}
- **Process Monitor** Processes running on user's computer
  Attributes {process id, user id, command, argument(s), Date Time, ...}

Figure 7.4: Lifetime Information Items in User's Repository

By applying our approach of *Connecting Ontology*, we can connect these two heterogeneous ontologies of *Projects* and *DataFeeds*. The following scenarios could be achieved:

a *"Find the data feed item related with project document"*

b *"Find the time invested in working on a specific project document"*

c *"Finding data feed items related with the first SemanticLIFE group meeting's participants"*

From the ontologies, we have the following information available:

- Project Ontology
  - Project document identification
  - Document authors
  - Modification Time

   - Location

- Data Feed Ontology
   - Information Item identification
   - Item's upload time
   - Location
   - Sender, Receiver
   - Process duration in case of process monitor data feed

Related with the previously described scenarios (a..c), we can state the rules as follows:

"If the upload time stamp of the data feed item is closer to the modification time stamp of the project document, then they may be related"

"If the human resource associated with the project document is equal to the user name in the process monitor data feed, then the process duration may be related with the project document modification time"

"If the upload time stamp of the data feed item is closer to the project meeting time, then the data feed items may be related with this specific project meeting"

The next step, in progress, is to convert these informal rule statements into *Semantic Web Rules*. That is to be done according to the classes and attributes of the concerned ontologies.

7.1.6 Summary

The automatic data capture devices like *SenseCam* do have the capability to capture much of user's activities. But, it is still far from building an automatic diary for the user due to missing associations. A proposal is suggested to capture the meeting snapshots and make those accessible for the blind. A *Project Meeting Ontology* is proposed which would be of particular benefit for blind people and generally beneficial for all. A number of rules in natural language are suggested which requires the connection between the *Project Meeting Ontology* and the *Data Feed Ontology*. The encoding of *Project Meeting Ontology* into *OWL* and the conversion and testing of informal rules into *Semantic Web Rules* is in progress using our approach of *Connecting Ontology*.

7.2 Towards Connection with Visualization Semantics

In order to build a representation with good usability and accessibility the various factors such as the type of information, the abilities of the users, the needs of the tasks and the device profile (as discussed in Chapter 2), must be taken into account. Here we argue that the *semantics of the information entity* in their own right, could also play a major role in suggesting the best visualization in prevalent situation. The different ontologies belonging to the different responsible factors must interact with each other. That can be done by using the *Connecting Ontology* approach described earlier.

The data about an "information entity" can be conceptualized in different ways according to the user needs and the task at hand. For example, the task objectives for an *Enterprise Resource Planning (ERP)* environments are different than those for *eLearning* environments. In an *ERP* environment there is a need for decision makers to monitor and track the product until it reaches the customer. The different phases start right from the preparation of customer oder, the bill of materials for preparing the production schedule, manufacturing and testing capacities of the enterprise, and thus meeting the orders of the customers and preparing invoices thereafter according to the materials consumed, man / machine time invested, and the transportation costs for the delivery of the product. On the other hand, the perspective for an *eLearning* environment can be completely different for the learners using the same data set. The point of interest might be to study the historical evolution of teaching stuff and the associated users impairments-related reasons for those evolutions.

Typically, the classical business intelligence visualizations would suffice for the first scenario, whereas those are of little use in the later one. Although in both the scenarios we will be making use of knowledge components of the involved knowledge workers. But the semantics to describe these components will be different because they serve different purposes in their respective domains. The domain concepts can be formally specified in ontologies. The challenge is to interpret the two different ontologies and determine what is being described. Once, that is done up to significant accuracy then appropriate visualizations can be suggested for that ontology. The whole process may be broadly distributed in three steps comprising the *schema development for user's information*, the *classification of information visualizations*, and the *connection between these two*.

The choice of visualization alternatives can then be performed automatically due to a matching mechanism dealing with the semantics for each type of visualization stored in the ontology. Thus the connection between the characteristics of the visualization metaphor and the semantics of information entity can be made on the fly. Then it could be possible to activate the most suitable visualization for the concept encountered while browsing the information space. For example, the query on class *Person* could be displayed as a typical *Contacts* look and feel, while a click on *Location* could activate geographical map visualization with the ability to switch back and forth between the two.

7.2.1 Background

A comprehensive taxonomy of visualization techniques is provided in (Chi, 2000). The taxonomy is based upon the *Data State Reference Model* which divides the task of visualization into four distinct stages. The functionality required for the transformation of data from one stage to another is termed as *operator*. The three *operators* or *transformations* are *"data transformation"* (performs primary analysis on the raw data to produce analytical abstraction or the meta-data), *"visualization transformation"* (reduces analytical abstraction to visualization abstraction or visualize able content), and *"visual mapping transformation"* (presents a graphical view from the normalizable content) where specialized techniques can be applied at each stage. The advantage is that the complex visualization task is broken down into multiple manageable sub-tasks. We can add another stage *"semantics extraction"* on top of visualization transformation. The job of this stage would be to read the concerned ontology and extract the domain knowledge which would be useful for visualization in rest of the stages. This could be an extension to the existing *Data State Reference Model*.

In (Winckler et al., 2004), tasks and scenario-based evaluation of information visualization techniques is proposed. It makes use of *Concurrent Task Trees* (Paterno et al., 1997) for task modeling and scenarios generation. The test is made for comparing the two hierarchical visualization techniques, namely the *TreeMaps* (Johnson and Shneiderman, 1991) and the *Hyperbolic browser* (Lamping et al., 1995). The suggested proposition asks for the automation of the task models and the scenario generation. We could extend it further to incorporate the automation aspect of affordances of different visualization techniques.

The taxonomy of tasks by data types provided in (Shneiderman, 1996) is a useful starting point for building more abstract domain related tasks. The stated data types are one, two, and three dimensional data, temporal and multi-dimensional data, tree and network data. The tasks are overview, zoom, filter, details on demand, relate, history and extract.

A problem-oriented classification of visualization techniques is described in (Wehrend and Lewis, 1990). The focus is on scientific visualization. However, the concept is also useful in the domain of information visualization. The classification suggests breaking a problem into sub-problems and then describes these using objects to be represented, operations to be performed on those objects, finding suitable visualization techniques, and combining these representations for solving the original problem. Objects are the visualization problems and the operations are corresponding visualization techniques. The examples of object classes are scalar, scalar field, nominal, direction, shape, position, spatially extended object and structure. The examples of operation classes are identify, locate, distinguish, categorize, cluster, distribution, rank, compare, associate and correlate. As discussed in (Morse et al., 2000), it is a low-level generalized tasks classification. However, the operation classes can also be used with application level object classes, like in case of personal information management tasks.

A comprehensive visual tasks categorization is described in (Zhou and Feiner, 1998). The categorization consists of the visual accomplishments (inform, enable), and visual implications (visual organization, visual signaling, visual transformation). The task "inform" is further categorized into elaborate and summarize tasks. Whereas, "enable" is divided into explore (search, verify) and compute (sum, differentiate). Visual organization is achieved through visual grouping (based upon proximity, similarity, continuity, closure (Mullet and Sano, 1995)), visual attention, visual sequence and visual composition. Visual signaling is divided into visual structuring and visual encoding. Visual transformation is divided into visual modification and visual transition. All of these can be composed by the low level visual task primitives such as associate, background, categorize, cluster, compare, correlate, distinguish, emphasize, generalize, identify, locate, rank, reveal, switch and encode. It is a very comprehensive categorization and it successfully demonstrates the visual discourse synthesis based upon above categorization.

Another level of abstraction which could be beneficial towards automation of *UI* and visualizations, is related with *tasks*. The *Concurrent Task Tree Environment (CTTE)* (Mori et al., 2002) specifies a notation for developing and analyzing the task models. The *CTTE* provides a *GUI* for creating and managing the task models for the developers and the designers. It also produces *XML* output for the task models. However it does not support the automatic generation of *UI* from these task models. A similar level of abstraction for the task models could be provided to the end users at run time with the help of ontologies.

In (Limbourg and Vanderdonckt, 2004) the mapping issues between different aspects of an interactive system during the development cycle are described. A model based approach to mapping has been discussed. The key issues described are domain specific usage for the definition of model and relationships, hard coding of model and its relationships, and inconsistent handling or treatment of relationships. Models cover the different aspects of the system such as task, domain, presentation, dialog, and context of use. Whereas, the context of use is formed by the user, the platform and the environment. It provides facility for mapping the whole source model into the target model or map selected elements between two models, and within one model itself. The models can be descriptive (used mainly for specifications purposes) or generative (used for generating some other information or code). It might work as follows:

A list of end user tasks is specified in natural language. Optionally, the recommended visualization for each task is also specified. The description of tasks in user's natural language and style would be very convenient for the user. Instead of recommending a sophisticated task syntax by the system the user would be able to formulate the tasks himself. Thus it would be possible for him to refine his list of tasks so as to optimize the task execution. However, in order to share the task lists by others it is imperative that either the tasks are specified in a universally agreed semantics or there exists a mapping between the two. Without compromising the user's convenience the suitable option is to provide a mapping. Each task is parsed and the keywords are mapped against available domain ontologies (for example mappings in *obj = 0* in the *Information Flow Framework* (SUOWG, 2006)). The visualization field would also be mapped appropriately in the ontology for visualization techniques. The process would be under user control so that he / she is able to confirm the mappings made. Next step is the transformation of list of tasks into appropriate task ontology. It is also noticeable that the data about user's information items and the tasks are not alike (Kelly et al., 2002). A fine line clearly separates the two. The data is considered to be facts which are represented fairly easily in the ontologies in terms of triples *(Subject, Predicate, Object)*. In *OWL*, facts can be described using the relationships such as *kindOf* and *partOf*. Whereas the tasks can be considered as imperative statements required to achieve a certain goal. Their representation in triples is non-trivial which needs further investigation of task models.

7.2.2 Overview of Ontology Interaction for Visualization Subsystem

The different factors mentioned earlier in the Section 7.2 must be integarted to contribute towards effective visualization of user's heterogeneous lifeime information. The integration of these different factors is to be carried out using our approach of *Connecting Ontology.* As a result, we will have more than one *Connecting Ontology* in our system. Since, we are here concerned with the visualization, therefore our intention is not to create a mesh of *Connecting Ontologies* inter connecting all the componets with each other. But to connect all the components with the visualization ontology only. Each of these resulting ontologies will describe the *suggestions* or *suitable matches* between the ontology of visualization techniques and the ontology of application component such as the device profile, the user's impairment profile and the user interface characteristics. For an overview of the situation see the Fig. 7.5).

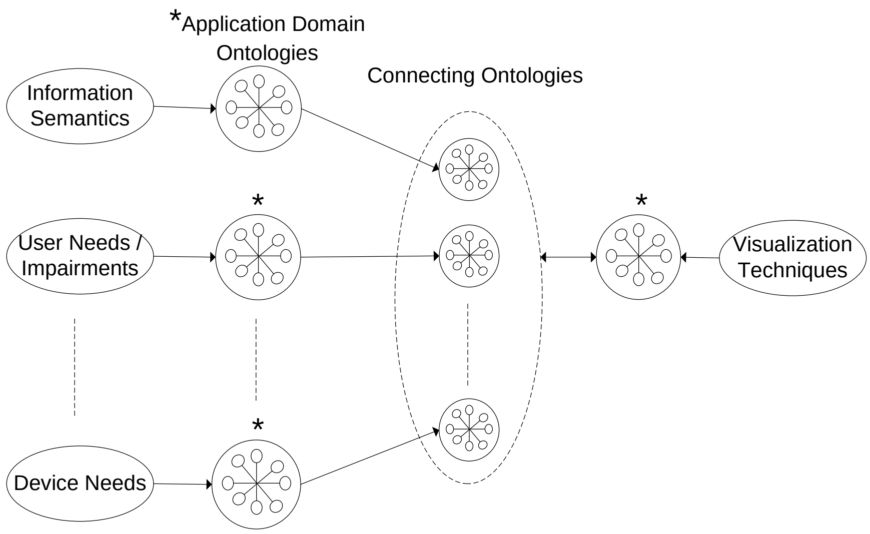

Figure 7.5: Overview of Ontology Interaction

In order to apply or use the *Connecting Ontologies* thus made, we must have a basic visualization system in place. This is described in the next sub section.

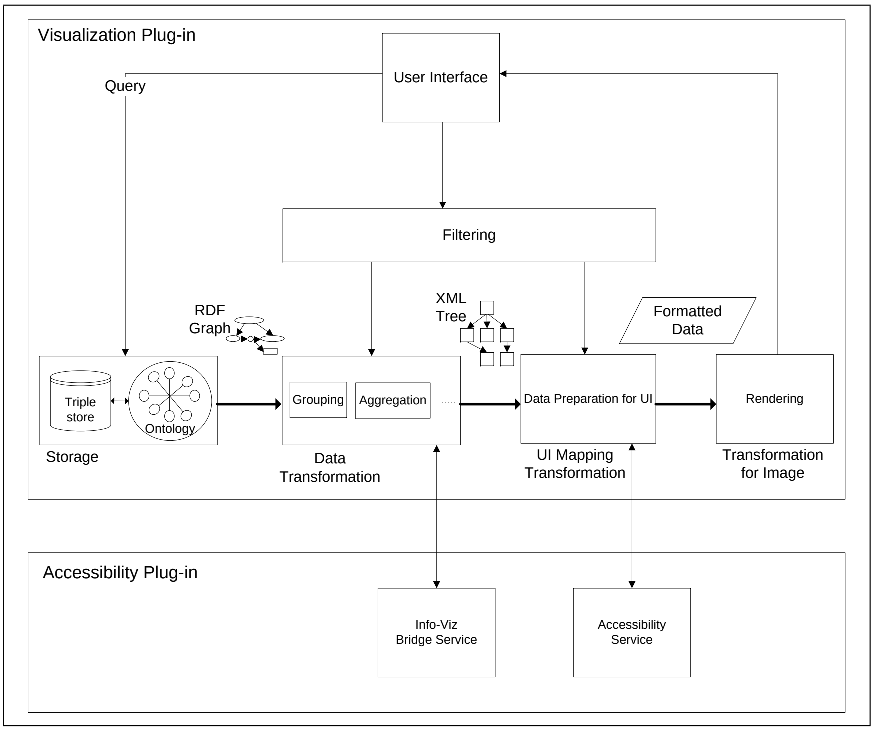

Figure 7.6: Overview of Visualization Subsystem

7.2.3 Proposed Approach for Visualization Subsystem

The goal is to provide multiple visualizations for viewing the heterogeneous information space. A number of visualization tools and techniques are described in (Card et al., 1999b). The visualization strategy is to provide an overview of the information first. Then the user has the ability to zoom in, with the possibility to filter the information according to different criteria. Lastly, from the filtered set of information items, the user can see the detailed contents of the items of his / her choice (Card et al., 1999a). Every visualization technique is not suitable for all of these visualization steps. This is a kind of *visualization pipeline*, which describes the steps from raw data to its visual representation for the end user.

It is implemented as a *Visualization Plug-in* in our system (see system overview in Fig. 2.4). The architectural overview of the *Visualization Plug-in* and its interconnection with the *Accessibility Framework* is given in Fig. 7.6. The details about the *Accessibility Framework* are already explained in Section 5.6.

Visualization Pipeline

The process of visualization is implemented as a sequence of following phases:

- **Storage**: Queries are sent by the client which are executed over the ontology of information items using *Jena Ontology API* (Reynolds, 2007). The store can be queried either programmatically or by sending *SPARQL* query strings. The result is sent as an *RDF Graph* to the *Data Transformation* module.

- **Data Transformation**: The data is transformed according to the requirements of the user. It consists of a sub-module for *Grouping* the data according to multiple criteria, and a sub-module for *Aggregating* the grouped data according to separate criteria. For our current visualization, the axis used for grouping and aggregation is based upon *time line* of information items. It would also be possible to use another conceptual axis other than the time, such as person, location and project. The *RDF Graph* is processed. The *XPATH* (Clark and DeRose, 1999) and the *XQUERY* (Boag et al., 2007) languages are used to do the transformation using the *XSLT* transformer (Clark, 1999). The *XML* output is passed to the *Mapping Transformation* module.

- **Mapping Transformation**: Its job is to transform the *XML* data according to the required format of the rendering engine. We used the *JFreeChart* (Gilbert, 2000) for implementing the *Time series* visualization, which requires a specific formatted text file.

- **Transformation for Image**: This step is hard coded and does not yet produce any structured output file. Because this is dependant upon the visualization toolkit being used. The output is the visualization of aggregated information items displayed along a time line, as shown in Fig. 7.7.

- **Filtering**: The user is able to control the filtering of items by selecting or deselecting the check boxes for each group of items. The request for filtering goes directly to *Data Transformation* or *Mapping Transformation* modules, depending upon the nature of requested filtering. The intermediate results are stored in persistent storage under user control, which can be used later in offline mode without launching queries over the *Storage*, thus improving the performance of the system.

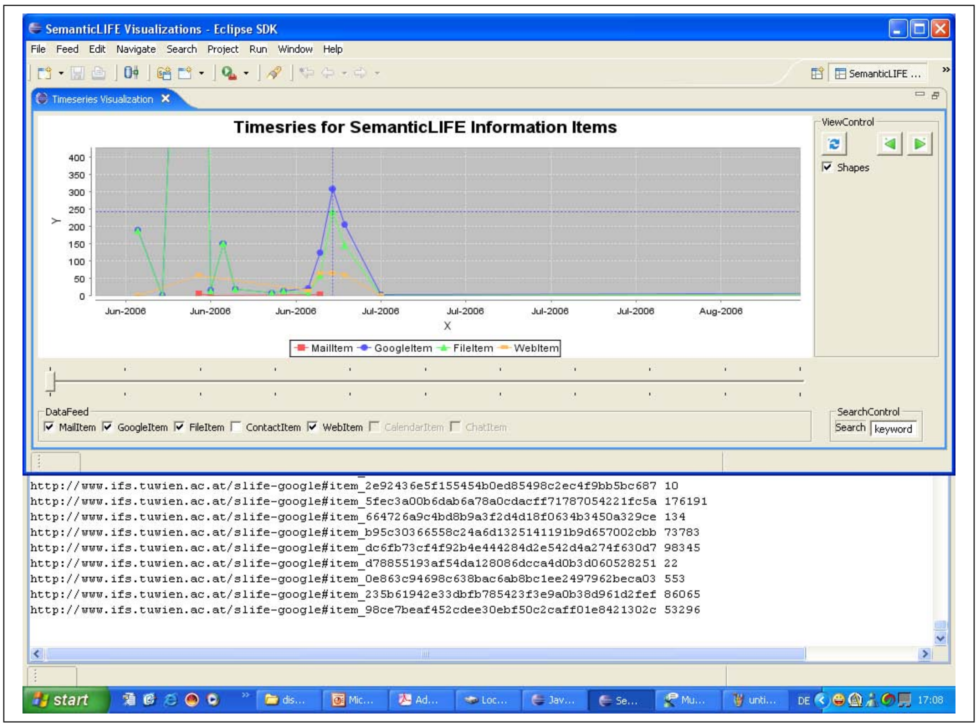

Figure 7.7: Grouping and Aggregation of Items Arranged on a Timeline

Information Overview

Since the number of information items gathered in user's lifetime personal store are huge, and can not be effectively visualized all at once. Therefore, we need intuitive visualization techniques to get an overview of the data. Visualizing the groups of data along the time line is implemented in our system, as shown in Fig. 7.7. It also provides the *zoom-in* and *zoom-out* facility.

Information Browsing

By clicking on the shape of grouped items in the visualization for information overview, all the constituent items are retrieved. Those are further viewable for browsing their detailed contents. The look and feel is as shown in Fig. 6.6. The user can navigate to the desired group of items. By clicking on the individual items, the contents can be seen.

Moreover, the *Browser* window is synchronized with the *Time series Visualization* window. The filtering done on either screen affects both.

For implementing the *Browser* interface, the output from *Data Transformation* module (see Fig. 7.6) is forwarded to the *Cascading Style sheet*

Adaptor. Its task is to adapt the data according to the user and device capabilities and preferences.

7.2.4 Summary

With the help of sample ontologies for two different domains we have emphasized how the semantics can be used to suggest suitable visualizations in each case. Also, while traversing ontology, visualization which goes well with the semantics of the current node, is more suitable instead of visualizing all the nodes with one technique. The work is still in progress. The visualization subsystem is implemented, with basic functionalities of visualization pipeline. A visualization for providing an information overview, and another for browsing the information items are implemented. Now we are in the process of classifying different visualization techniques so that the most suitable one is chosen on the fly based upon the semantics of the contextual elements with the the help of *Connecting Ontology* approach.

Chapter 8

Discussion and Conclusion

Providing a *fit for all* or a generic accessibility solution for different types of impairments, is not an easy goal to achieve. Rather it is somewhat unrealistic because different users have different needs, capabilities and impairments which are difficult to incorporate giving due weightage to each for every user. However, by using our proposed *Connecting Ontology* approach it is possible to build a generic foundation which can further be extended for specific special needs of the user.

8.1 Contributions

The significant contributions made in this thesis are as follows:

- Ontologies for modeling the impairments of the user, called the *Impairments Ontology* and for modeling the user interface characteristics, called the *User Interface Ontology* are proposed. It is demonstrated how the impairments and the user interface data can be encoded in *OWL DL* which allows inferencing. These ontologies are connected with each other for adapting the user interface for specific needs of the user.

- A number of *Conceptual Ontology Design Patterns (CODeP) for Accessibility* are introduced. This starts with a generic *CODeP for Accessibility* which can further be extended or specialized. Following these lines, a *CODeP* for *Perception Effect Pattern* is introduced which depicts the *cause-effect* relationship between the user impairments, user interface characteristics and the device profile. A *Memory Recall Pattern* is introduced which is helpful for users with memory problems. It describes the relationship between the user's recall ability, his lifetime information which is associated with life events, and the representations to present those events to the user. The last one is the *Mobility Enhancement Pattern* for helping the users with mobility related problems. It describes the relationship between the user's mobility problems, his lifetime information which is associated with different mobility cues, and the representations to present the query results to the user.

- The concept of *Connecting Ontology* is introduced to connect any two heterogeneous information domains. This is demonstrated using a test case of the *User's Impairments Ontology* and the *User Interface Ontology*. The two *OWL DL* ontologies are connected using the *Semantic Web Rules* resulting in another ontology. However, the new ontology, called the *Connecting Ontology* is in *OWL Full* and may need to be transformed in *OWL DL* for meeting specific requirements.

- The thesis also demonstrates how the rule layer in the *Semantic Web Architecture* can be implemented for developing more higher level ontologies.

- The adopted approach is useful in integration of information systems and in knowledge management. It is also to note that the method to connect two heterogeneous domains is helpful in explicitly specifying the tacit knowledge of the users, which is a step towards new knowledge creation. Since the approach is desirous of having autonomous ontologies of two domains instead of mixing the semantics in a single ontology, therefore the synthesis of knowledge in specialized domains is encouraged. This is a step forward in avoiding the hard coding of tacit knowledge, and thus helpful in code automation.

- A few motivating scenarios are described in detail where our approach in context of the personal information management system *SemanticLIFE* could possibly be employed.

  The *SenseCam Scenario* indicates the use of our system for providing accessibility in carrying user's general activities which are not strictly computer based.

  The *Visualization Scenario* indicates how the approach could be used during different phases in the visualization pipeline. This is helpful in visualizing the information items on multiple axes. In the implemented visualization subsystem, the information items are visualized along time line on horizontal axis. The grouping of items on vertical axis is based upon the number of information items uploaded in the particular time slice which can be dynamically adjusted by the user.

8.2 Future Directions

The effort in the thesis was also fruitful in the sense that many future research directions are anticipated. Some of those are briefly described here:

- **User Testing:** The feedback from the users is necessary to validate the results and for making further improvements. We plan to test our system with normal users as well as with users with special needs.

- **Semantic Web Services for the *Accessibility* and *Info-Viz Bridge* Modules:** These two modules of the *Accessibility Framework* would be converted to *Semantic Web Services* so that these are usable by the *Agents* over the *Web*.

- **Capability Calibration Service:** This service will be responsible for conducting user controlled testing to judge user's capabilities. Based upon the results, the impairment ontology will be instantiated regarding the perception cues and the capabilities. There exist some initial tests like *Mini Mental State Examination(MMSE) for Diagnosis of Dementia*[1], and an automated tool for *MMSE*[2]. The later maintains a longitudinal profile of the patients over time. Our idea is to exploit the *SemanticLIFE* system, for automatically carrying out these tests. For example, based upon the picture annotations, useful scenarios can be developed which could substitute the testings done in *MMSE*.

- **Semantic Web Services for the Ontologies:** This includes the deployment of Semantic Web Services for the *Impairments Ontology* and the *User Interface Ontology*. Moreover, the terms in the *Impairments Ontology* are to be aligned with the standard ontologies in medical domain such as the *Human Disease Ontology* and vocabularies like *ICD9CM* (OBO, 2007).

- **Other Applications of Impairments and UI Ontologies:** After a certain level of maturity, these ontologies could also be very useful in other domains. For example, the *Impairments Ontology* could be used for *eLearning* in medical domain. Whereas, the *User Interface Ontology* could be used in the domains of *eLearning* and *Usability Engineering*.

- **Realization of Remaining *CODeP for Accessibility*:** In the thesis the *Perception Effect Pattern* is implemented. Similarly other patterns mentioned in Chapter 4 can be realized. Effort is under way to

[1] http://www.alzheimers.org.uk/How_is_dementia_diagnosed/Diagnosis_process/info_mmse.htm (5th September 2007)

[2] http://www.minimental.com/MSRS.htm (5th September 2007)

implement the *Memory recall Pattern* as part of the *SenseCam Scenario* described in Section 7.1. Similarly, the realization of *MobilityEnhancementPattern* could be very useful in context of various applications such as the adaptation of the *Google Earth* [3], for taking into account the mobility related constraints of the user.

- **Ontology for Visualization Techniques:** The semantic representation of different visualization techniques so that the visualizations are not chosen in advance, but could be available for on-line selection by the user depending upon the prevalent information semantics during navigation and browsing.

- **Ontology Elevation:** Based upon the execution of rules between two ontologies (*Impairments Data* and *User Interface Characteristics*), a new ontology (*Connected Ontology*) is formed. This ontology is able to fulfill certain competency questions. Similarly, the *User's Impairments* domain can be connected with a domain other than the *User Interface Characteristics* for making another *Connecting Ontology.* Now these two *Connecting Ontologies* can further be connected to accomplish a more high level scenario.

8.3 Evaluation of the Goals

At the beginning of the thesis, our first research question was:

> *Can Semantic Web Technology be used for providing a Generic Accessibility solution base for people with special needs?*

Looking at the *Semantic Web Layered Architecture* in Fig. 2.3 the tools and techniques falling under its umbrella range from *XML Technologies* of *Web 1.0* for data representation, the sophisticated tools and techniques specific to *Semantic Web* for representing the information semantics, and the associated *APIs* for programmatic interaction. The method adapted in our approach used many tools and techniques specific to different layers. For example, the encoding of the two heterogeneous domain ontologies was done in *OWL DL*, their interconnection was accomplished using *Semantic Web Rules* and *Jena Ontology API.* Then the results were post-processed by the *XML Technology.* The transformation of the fetched data was done by *XSLT* transformations which used *XPATH* and *XQUERY* statements in the *XSL*

[3]http://earth.google.com/ (5th September 2007)

files. For rendering the results in the browser like *Mozilla Firefox* and *Internet Explorer*, the *XSL* stylesheets and the *CSSOM parser* were used. In this thesis the user interface adaptation based upon the user's impairments and the user interface characteristics is completely accomplished using the *Semantic Web Technologies* mentioned above. However, our experience is that the most popular visualization toolkits are not yet ready to work with the *Semantic Web Technology* from the point of view of rendering of the visualization. It is normally hard coded into the concerned visualization *API*. Another aspect is that the representation of semantics of different impairment types is a very complex task, but an essential step towards automation. Unless it is accomplished, the claim that our approach provides a generic accessibility solution is a little bit too optimistic. Nonetheless, it provides a generic accessibility foundation which may further be customized for specific types of impairments.

The second research question was:

> *How far is it justified to assume that the investment on UI alone could provide maximum possible Accessibility for people with special needs?*

The work about the motivating scenarios in Chapter 7 shows how information semantics can be organized around life events, gestures during the meetings and the projects. However, the complete results for these scenarios are not yet available. In perspective of different ontology design patterns defined in Chapter 4, the search and display for the different types of information semantics may vary depending upon the impairments of the user. Moreover, the semantics are enhanced in terms of annotations in *OWL, RDFS* which can be specific to the prevalent user's context. In that case, it is fair to say that accessibility can not be sufficiently enhanced by only investing on user interface layer in the *Semantic Web Layered Architecture* (Fig. 2.3). The time line of the *Semantic Web Layered Architecture* literature indicates that the issues of *User Interface*, *Usability* and *Accessibility* are very recent. Based upon the discussion made in this thesis, we believe that the *Usability* and *Accessibility* layer stands vertical in the *Semantic Web Architecture* (see Fig. 8.1).

The last research question was:

> *Is the sought-after approach exploitable towards diversity in general, and integration of Information Management Systems in particular?*

The demonstration is made for the *User's Impairments* and the *User Interface Characteristics*. However, the adapted approach to connect two

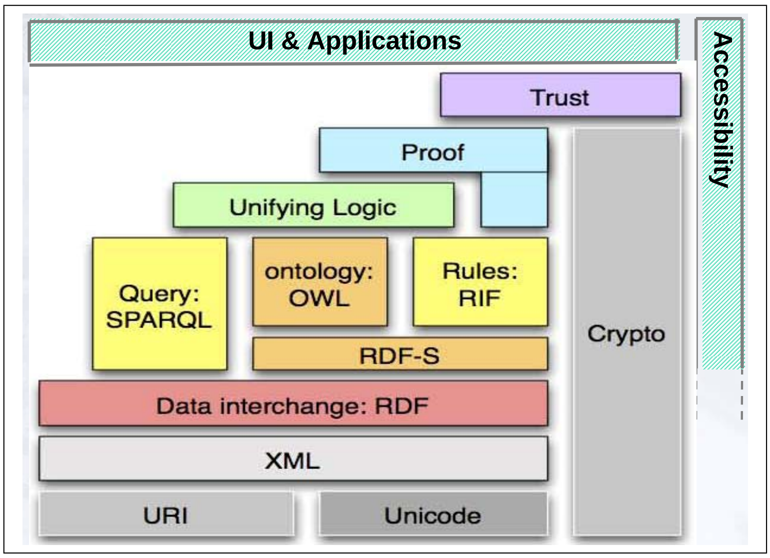

Figure 8.1: Semantic Web - The Modified Architecture

heterogeneous domains using domain specific rules is equally applicable for any domain of discourse. Certainly, the key issue is to make the rules which will be largely dependant upon the domain experts and other innovative data mining techniques for searching knowledge about the rules. Therefore, in principle the approach is also good for the integration of *Information Management Systems* which are composed of many heterogeneous sub-domains. The modeling of user specific attributes such as the impairments would definitely help to promote *Diversity* of the *Web*.

8.4 Summary

In this thesis we have described an approach using the *Semantic Web Technology* which can be used to connect ontologies belonging to heterogeneous information domains. The approach is demonstrated using the test case of connection between the *User's Impairments Data* and the *User Interface Characteristics* for automatic interface adaptation. This has significant benefits for providing *Accessibility for People with Special Needs* thus improving their productivity while working with computers. Since the impairments are considered as one of many contextual components in a system. Therefore, by following the approach which is useful for information integration of these contextual components, we have introduced the impairments domain in the mainstream too. This has paved the way for a common ground where the efforts done for the people with special needs and people without any special needs, will be fruitful for both.

The importance of modeling the *User's Impairments Data* and the *User Interface Characteristics* as *OWL DL* ontologies, is also highlighted. We have also described some motivating application scenarios where our approach can be successfully applied.

Appendix A

Questionnaire For Task Analysis

Your Name:____
Date:____________

Research Area:

SemanticLIFE Component (such as annotation / query / ontology management / user profile / dynamic associations / privacy and access control):

Reference Scenario: *"Researcher Working on Multiple Projects"*
==
The following list (tasks and information objects) is numbered according to their importance in our system.

1. Characteristics of task

   (a) Name the tasks or operations to be performed by this component. For example, annotate one class to another class; annotate one instance to another instance. In case of difficulty in identifying the task details, one of the good strategies suggested by experts is to "think aloud", i.e., speak out or write down the task you are intending to do.

   (b) *Mandatory or Non-Mandatory tasks*; By mandatory tasks we mean those tasks without which the component is considered to be Non-Functional?

   (c) *Task frequency*; how many times per day, per hour, etc

   (d) *On line or batch execution*; Mark the task as executing on line, or in batch mode. For example, by seeing an email item we may try to annotate it instantly, instead of dragging and dropping many email items on *Annotation* icon, and then do the annotation sometime later.
   (e) *Task priority*; such as High, Medium, Low. High priority tasks can be very crucial and urgent for the user.
   (f) *Task nature*; Is it a management task such as saving, importing annotation, or core functionality like doing the annotation itself?
   (g) *Task dependency*; Describe dependency on other task(s) maybe in other components
   (h) Any other specificity about the nature of the task

2. Characteristics of information object
   (a) *Type of information object*; the concerned information objects can be class, instance, property or the ontology. For example, the named queries can also be treated as first class objects for query.
   (b) *Nature*; such as an email, web page, contact, picture, blob.
   (c) *Size*; the estimated size will vary depending upon the concerned information objects for your component
   (d) *Frequency*; Try to give an estimate about the frequency of incoming information objects to your component
   (e) Any other specificity about nature of information items

3. *Filtering / Control options*; What are the options for user control? For example to filter the items according to date, popularity, or some other criteria. Describe the result fields after applying filtering, if possible.

4. *Tracking of task state*; One task may consist of several steps. Is it beneficial to track those steps for replaying later?

5. *Status messages*; Is it necessary to display the status message for this task? If yes, what status information is to be displayed?

6. *Visible / Invisible navigation*; Although a task can be made automatic as far as possible, however, it is sometimes beneficial to display the task execution steps to the user for developing a mental image

7. *Reversible actions*; Like undo functionality. During execution of a sequence of queries, is it beneficial to undo the selected queries instead of undoing all.

8. *Copy / Paste functionality*; For example, apply one type of recently made annotation on another collection of items. Or copy one sub-query from a named query to another named query.

9. *Scratch pad functionality*; Is it useful for the task to drop the items upon scratch pad, for taking action later?

10. *Active items list functionality*; Is it useful for the task to have frequently or recently addressed items on the Active Items list

11. What should be the default view in your opinion?

12. Any other thing you feel necessary to describe

Note: We also welcome your feedback about any other component of your interest, other than the component you are working on.

Appendix B

Ontology Development Template

```
1. Define General Ontology Characteristics
     Purpose and Scope
     Interaction With Other Ontologies
     Intended Ontology Users
     Usage Scenarios and Competency Questions
     Formalism

2. Design Ontology Contents
     Collection of Terms / Concepts
     Identify Primitive and Derivable Concepts
     Organize Taxonomy
     Find Relationship of Concepts

3. Describe Derivable Concepts
     Restrictions
     Rules

4. Define Ontology Population
     Identify the Data Source
     Identify the Data Characteristics (such as size, type,
                             frequency, archival time)
     Identify the Data Capture Mechanism
     Identify the number of triples per data units
     Identify the number of triples per concept in the ontology
     Plan the Population Mode (on line or periodically in batch)
```

```
5. Evaluation
     Validate Competency Questions
     Validate Scenarios

6. Revise the Steps as required
```

Appendix C

Taxonomy of User Impairments

The word *Impairment* has different meanings in literature. For example, according to the *WordNet* (Fellbaum, 1998) some of the senses are:

- the occurrence of a change for the worse
- a symptom of reduced quality or strength
- the condition of being unable to perform as a consequence of physical or mental unfitness
- damage that results in a reduction of strength or quality
- the act of making something futile and useless (as by routine)

If the concept of *Impairment* is related to the person then it is much closer to the *Disease* as in the *Human Disease Ontology* (OBO, 2007). However, as is evident from the above senses, the concept may also be related with the environment or the context. In that case it is like an *Interaction Constraint* for the user (Obrenovic et al., 2007). Both the concepts can be used interchangeably depending upon the situation. However, for the purpose of *Impairments Ontology* in this thesis, our intention is to adopt the former concept, i.e; related to the person.

The *Impairment* may be *Inherited (genetically transmitted)* or *Acquired (due to environmental forces).* The user with disability or impairment may be dependant or reliant upon someone for carrying out his / her activities. The reliance can be nil, partial or full, depending upon the severity of the impairment. There can be various types of impairments. A representative sample from those is categorized in groups as follows:

Note: The given impairments taxonomy is not complete. The intention is to model a few impairments so as to demonstrate the working of our system. However, it can be enhanced using the knowledge available from ontologies in medical domain as indicated in Chapter 6.

1. Physical Impairment (general loss of physical ability)

    - Vision, causedBy Eye
        - ColorBlindness (Inability to perceive some or all colors)
            - Red-Green ColorBlindness
                - Protanopia (unable to distinguish green-yellow-red section, rare)
                - Protanomaly (less sensitive to red light than normal, red confuses with black, rare)
                - Deuteranomaly (less sensitive to green light than normal, most common)
            - Blue-Yellow color blindness
            - Monochromacy (complete inability to distinguish any color)
                - Cone Monochromacy (vision is normal)
                - Rod Monochromacy (vision problem in lights of normal intensity)
        - Blindness (Lack of visual perception)
            - Monocular (affecting one eye) / Binocular (affecting both eyes)
            - LegalBlindness (visual acuity of 20/200 or less; person is allowed to avail blindness privileges as allowed by law)
            - CompleteBlindness (inability to perceive light)
            - Hemianopsia (absence of vision in half of visual field, each eye misses a half circle of visual field, left or right sided)
                - HomonymousHemianopsia (a half of visual field is missing on same side, right or left)
                - BitemporalHemianopsia (outer 1/2 visual field missing on both sides)
                - BinasalHemianopsia (inner 1/2 of visual field missing on both sides)
                - Quadranotopia (similar to Hemianopsia above with same types, except that the affected visual field is 1/4th instead of 1/2)

    * Scotoma (An island of loss of vision surrounded by normal vision, black hole in centre)
    * Diplobia (seeing double, in all field of vision or only in left/right eye)
    * Hemeralopia (day blindness, partial or complete loss of vision in bright light or day light)
    * Nyctalopia (night blindness, partial or complete loss of vision during dim light or darkness)
    * TunnelVision (constricted field of vision in both eyes)

- Vision, causedBy Brain
  - Agnosia (loss of ability to recognize objects, people, sounds, shapes; the inability to attach appropriate meaning to objective sense-data)
    * VisualAgnosia, synonym-CorticalBlindness (blindness due to processing part of the brain, brain looses the capacity to make sense of the image presented, could be due to stroke or tumor)
    * ProsopAgnosia (inability to recognize familiar faces at times their own)
    * ColorAgnosia (inability to recognize one or more colors, in spite of normal eyes and anterior visual pathways)
    * TunnelVision (constricted field of vision in both eyes, due to migraine or in psychiatry)
- Mobility
  - Permanent / Temporary
  - LowerBody (Canes, Walkers, WheelChairs) / UpperBody (Limited or no use of the upper extremities and hands)
  - Orthopedic / Neuromuscular

2. Cognitive Impairment (general loss of mental or cognitive ability, related with lack of perception/ learning/reasoning)

   - Aphasia (inability to use or understand spoken or written language)
     - Broca (Motor aphasia, spontaneous speech and fluency diminished, cannot name object, comprehension normal or mildly affected, repetition may or may not be affected, reading affected, writing poor, condition varies from mutism to hesitant speech)

    - Wernicek (sensory aphasia, main defect is of comprehension of spoken language, reading and writing. Patients own spoken language is fluent though not making any sense, repetition and naming is also poor)
    - Global (mixture of above two conditions, unable to understand or produce any spoken or written language)
    - Gerstmann syndrome (Agraphia, inability to calculate, right-left confusion, and finger agnosia - inability to recognize fingers)
- Dyslexia (learning disability that affects language processing and reading skills, average or above-average intelligent people)

Appendix D

List of Publications

During the course of my PhD studies, the following papers were published.

- Karim, S. and Tjoa, A. M. (2007). Connecting User Interfaces and User Impairments for Semantically Optimized Information Flow in Hospital Information Systems. In Print, Journal of Universal Computer Science: Proceedings of I-MEDIA'07 and I-SEMANTICS'07, pages 372-379.

- Karim, S., Latif, K., and Tjoa, A. M. (2007). Providing Universal Accessibility using Connecting Ontologies: A Holistic Approach. In Constantine Stephanidis, editor, Universal Access to Applications and Services, volume 7 of LNCS 4556, pages 1147-1154, Springer - Berlin / Heidelberg.

- Karim, S. and Tjoa, A. M. (2006). Towards the Use of Ontologies for Improving the User Interaction for People With Special Needs. In Proceedings of the 10th International Conference on Computers Helping People with Special Needs ICCHP'04, volume 4061 of LNCS, pages 77-84, Springer - Berlin / Heidelberg.

- Karim, S. and Tjoa, A. M. (2006). Exploiting SenseCam for Helping the Blind in Business Negotiations. In Proceedings of the 10th International Conference on Computers Helping People with Special Needs ICCHP'04, volume 4061 of LNCS, pages 1147-1154, Springer - Berlin / Heidelberg.

- Anjomshoaa, A., Karim, S., Shayeganfar, F., and Tjoa, A. M. (2006). Exploitation of Semantic Web Technology in ERP Systems. In Tjoa, A. M., Xu, L., and Chaudhry, S., editors, Research and Practical Issues of Enterprise Information Systems, volume 205 of IFIP International

Federation for Information Processing Series, pages 417-427, Springer Boston.

- Weippl E., Schatten A., Karim. S. and Tjoa., A. M. (2004). 'SemanticLIFE Collaboration: Security Requirements and Solutions - Security Aspects of Semantic Knowledge Management. In Proceedings of the 5th International Conference on Practical Aspects of Knowledge Management, PAKM'04, volume 3336 of LNCS, pages 365-377. Springer-Verlag.

- Ahmed, M., Hanh, H. H., Karim, S., Khusro, S., Lanzenberger, M., Latif, K., Elka, M., Mustofa, K., Tinh, N. H., Rauber, A., Schatten, A., Tho, N. M., and Tjoa, A. M. (2004). SemanticLIFE - A Framework for Managing Information of A Human Lifetime. Austrian Computer Society Book Series: iiWAS'2004,183:687-696.

Invited Talk:

- Providing Universal Accessibility using Semantic Web Technology', Presentation given at the Institute of Medical Informatics, Statistics and Documentation (Medical University, Graz), 20th June 2007, Graz - Austria.

Bibliography

Abascal-Gonzalez, J., Arrue-Recondo, M., Garay-Vitoria, N., Tomas-Guerra, J., and Velasco-Nunez, C. A. (2003). USERfit Tool: A Design Tool Oriented towards Accessibility and Usability. *UPGRADE The European Journal for the Informatics Professional Human-Computer Interaction: Overcoming Barriers*, IV(1):5–11.

acemeDia (2004). M-OntoMat-Annotizer. http://www.acemedia.org/aceMedia/results/software/m-ontomat-annotizer.html (5th Sep'07).

Ahmed, M., Hanh, H. H., Karim, S., Khusro, S., Lanzenberger, M., Latif, K., Elka, M., Mustofa, K., Tinh, N. H., Rauber, A., Schatten, A., Tho, N. M., and Tjoa, A. M. (2004). SemanticLIFE - A Framework for Managing Information of A Human Lifetime. *Austrian Computer Society Book Series: iiWAS'2004*, 183:687–696.

AMI (2007). Augmented Multi-Party Interaction. http://www.amiproject.org/ (5th Sep'07).

Angele, J., Boley, H., de Bruijn, J., Fensel, D., Hitzler, P., Kifer, M., Krummenacher, R., Lausen, H., Polleres, A., and Studer, R. (2005). Web Rule Language (WRL). Member submission, W3C.

Angele, J. and Gesmann, M. (2006). Data Integration using Semantic Technology: A use case. In Thomas Eiter and Enrico Franconi and Ralph Hodgson and Susie Stephens, editor, *RuleML*, pages 58–66. IEEE Computer Society.

Anjomshoaa, A., Nguyen, T. M., Shayeganfar, F., and Tjoa, A. M. (2006). Web Service Based Business Processes Automation Using Semantic Personal Information Management Systems The Semantic Life Case. In *Proceedings of Practical Aspects of Knowledge Management*, volume 4333) of *LNCS*, pages 1–12, Vienna, Austria. Springer - Berlin / Heidelberg.

ANSI (1997). ANSI / HFES 200 Draft. http://acm.org/sigchi/bulletin/1997.2/standards.html#HDR3 (5th Sep'07).

Baader, F., Lutz, C., Sturm, H., and Wolter, F. (2002). Fusions of Description Logics and Abstract Description Systems. *Journal of Artificial Intelligence Research (JAIR)*, 16:1–58.

Battle, S., Bernstein, A., Boley, H., Grosof, B., Gruninger, M., Hull, R., Kifer, M., Martin, D., McIlraith, S., McGuinness, D., Su, J., and Tabet, S. (2005). Semantic Web Services Language (SWSL). Member submission, W3C.

Baumgartner, R., Flesca, S., and Gottlob, G. (2001). Visual Web Information Extraction with Lixto. In *VLDB '01: Proceedings of the 27th International Conference on Very Large Data Bases*, pages 119–128, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Bechhofer, S., van Frank Harmelen, Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., and Stein, L. A. (2004). OWL Web Ontology Language Reference . Recommendation, W3C. http://www.w3.org/TR/owl-ref/.

Beckett, D. and ed., B. M. (2004). RDF/XML Syntax Specification. Recommendation, W3C.

Berners-Lee, T. (1999). *Weaving the Web. The original design and ultimate destiny of the World Wide Web by its inventor*. Harper, San Francisco.

Berners-Lee T., F. R. and L., M. (1998). RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax.

Bevan, N. and Macleod, M. (1994). Usability Measurement in Context. *Behaviour and Information Technology*, 13(1):132–145.

Bizer, C., Cyganiak, R., Garbers, J., and Maresch, O. (2006). D2RQ V0.5 - Treating Non-RDF Databases as Virtual RDF Graphs. User manual, Institut für Produktion, Wirtschaftsinformatik und OR - Freie Universität Berlin. http://www.wiwiss.fu-berlin.de/suhl/bizer/d2rq/spec/20061030/.

Boag, S., Chamberlin, D., Fernández, M. F., Florescu, D., Robie, J., and Siméon, J. (2007). XQuery 1.0: An XML Query Language. Recommendation, W3C.

Bos, B., Lie, H. W., Lilley, C., and Jacobs, I. (1998). Cascading Style Sheets, level 2 CSS2 Specification, W3C Recommendation. W3C Recommendation, W3C.

Bouquet, P., Giunchiglia, F., van Harmelen, F., Serafini, L., and Stuckenschmidt, H. (2004). Contextualizing ontologies. *Journal of Web Semantics*, 1(4):325–343. Web Semantics: Science, Services and Agents on the World Wide Web.

Card, S. K., Mackinlay, J. D., and Shneiderman, B. (1999a). Overview + detail. pages 285–286.

Card, S. K., Mackinlay, J. D., and Shneiderman, B. (1999b). *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann

Publishers Inc., San Francisco, CA, USA.

Carruth, D., Thomas, M., Robbins, B., and Morais, A. (2007). Integrating Perception, Cognition and Action for Digital Human Modeling. In Vincent G. Duffy, editor, *Digital Human Modeling*, volume 12 of *LNCS 4561*. Springer - Berlin / Heidelberg.

Chen, H., Colaert, D., and Roo, J. D. (2004). Towards Adaptable Clinical Pathway Using Semantic Web technology. In *W3C Workshop Semantic Web for Life Science*. Position paper for W3C workshop Semantic Web for Life Science.

Chi, E. (2000). A taxonomy of visualization techniques using the data state reference model. In *INFOVIS*, pages 69–75.

Chisholm, W., Vanderheiden, G., and Jacobs, I. (1999). Web Content Accessibility Guidelines 1.0. Recommendation, W3C.

Clark, J. (1999). XSL Transformations (XSLT), Version 1.0. Recommendation, W3C.

Clark, J. and DeRose, S. (1999). XML Path Language (XPath), Version 1.0. Recommendation, W3C.

Cunningham, H., Maynard, D., Bontcheva, K., and Tablan, V. (2002). GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*.

Dix, A., Finlay, J., Abowd, G., and Beale, R. (1993). *Human-Computer Interaction*. Prentice-Hall, New Jersey.

DOAP (2006). Description of a Project, The Useful Information Company. http://usefulinc.com/doap/ (5th Sep'07).

DynamOnt (2007). Dynamic ontology creation. http://dynamont.factlink.net/ (5th Sep'07).

Dzaack, J. and Urbas, L. (2007). Cognitive Model Data Analysis for the Evaluation of Human Computer Interaction. In Harris, D., editor, *Engineering Psychology and Cognitive Ergonomics*, volume 4562 of *LNAI*, pages 477–486. Springer.

Encelle, B. and Baptiste-Jessel, N. (2004). Adapting Presentation and Interaction with XML Documents to User Preferences. In *Proceedings of the 9th International Conference on Computers Helping People with Special Needs ICCHP04, LNCS*, volume 3118, pages 143–150. Springer-Verlag.

EU (2002). Accessibility of Public Websites – Accessibility for People with Disabilities: Council Resolution. http://www.legi-internet.ro/index.php?id=149 (5th Sep'07).

Fellbaum, C. (1998). *WordNet : an Electronic Lexical Database. Language, speech, and communication*. MIT Press, Cambridge, Mass.

Ferrario, R. and Oltramari, A. (2004). Towards a Computational Ontology of Mind. In A.C. Varzi, L. V., editor, *Proceedings of FOIS 2004*, pages 287–297, Torino, Italy. IOS Press.

Fielding, R. T. and Taylor, R. N. (2002). Principled Design of the Modern Web Architecture. *ACM Trans. Inter. Tech.*, 2(2):115–150.

Flickr2RDF (2007). Flickr Photo info to RDF Image Description. http://purl.org/net/kanzaki/flickr2rdf (5th Sep'07).

Fowler, A. (1998). A Swing Architecture Overview. http://java.sun.com/products/jfc/tsc/articles/architecture/ (5th Sep'07).

Gangemi, A. (2005). Ontology Design Patterns for Semantic Web Content. In Gil, Y., editor, *Proceedings of International Semantic Web Conference (ISWC)*, volume 3729 of *Lecture Notes in Computer Science*, pages 262–276, Galway, Ireland. Springer.

Gangemi, A., Navigli, R., and Velardi, P. (2003). The OntoWordNet Project: Extension and Axiomatization of Conceptual Relations in WordNet. In *CoopIS/DOA/ODBASE*, pages 820–838. http://www.loa-cnr.it/ontologies/OWN/OWN.owl.

Garcia, E. and Sicilia, M.-A. (2003). User Interface Tactics in Ontology-Based Information Seeking. In *PsychNology Journal*, volume 1(3), pages 242–255. PsychNology Journal.

Gemmell, J., Bell, G., and Lueder, R. (2006). MyLifeBits: A Personal Database for Everything. *Commun. ACM*, 49(1):88–95.

Gilbert, D. (2000). JFreeChart: A free Java chart library. http://www.jfree.org/index.html (5th Sep'07).

GNOME (2006). GNOME Usability. http://developer.gnome.org/projects/gup/ (5th Sep'07).

Grail (2002). Grail: Digital humans. http://grail.cs.washington.edu/projects/digital-human/ (5th Sep'07).

Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow, F., and Pirahesh, H. (1997). Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals. *J. Data Mining and Knowledge Discovery*, 1(1):29–53.

Gruber, T. (1995). Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *Human-Computer Studies*, 43(5-6):907–928.

Hadzic, M. and Chang, E. (2005). Ontology-Based Support for Human Disease Study. In *HICSS'05: Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 6*, page 143.1, Washington, DC, USA. IEEE Computer Society.

Haller, A., Oren, E., and Kotinurmi, P. (2006). m3po: An Ontology to Relate Choreographies to Workflow Models. In *SCC '06: Proceedings of the IEEE International Conference on Services Computing (SCC'06)*, pages 19–27, Washington, DC, USA. IEEE Computer Society.

Harper, S. and Bechhofer, S. (2005). Semantic Triage for Increased Accessibility. *IBM Systems Journal*, 44(3):637–648.

HCLSIG (2004). Semantic Web Health Care and Life Sciences Interest Group. http://www.w3.org/2001/sw/hcls/ (5th Sep'07).

Holzinger, A. (2005). Usability Engineering for Software Developers. *Communications of the ACM*, 48(1):71–74.

Holzinger, A., Geierhofer, R., and Errath, M. (2007). Semantische Informationsextraktion in medizinischen Informationssystemen. *Informatik Spektrum*, 30(2):69–78.

Hornbík, K. (2006). Current Practice in Measuring Usability: Challenges to Usability Studies and Research. *International Journal of Human-Computer Studies*, 64(2):79–102.

Horrocks, I. and Patel-Schneider, P. F. (2006). OWL 1.1 Web Ontology Language Overview. W3c member submission, W3C. http://www.w3.org/Submission/owl11-overview/.

Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B., and Dean, M. (2004). SWRL: A Semantic Web Rule Language Combining OWL and RuleML. Member submission, W3C.

Howard D. Wactlar, Michael Christel, A. H. S. S. (2003). A System of Video Information Capture, Indexing and Retrieval for Interpreting Human Activity. In *IEEE International Symposium on Image and Signal Processing and Analysis (ISPA'03), Special Session on System Perspectives in Information Retrieval*, volume), Rome, Italy.

ICSI (2000). The Meeting Recorder Project.
http://www.icsi.berkeley.edu/Speech/mr/ (5th Sep'07).

IM2 (2007). Interactive Multimodal Information Management, NCCR (SNSF). http://www.im2.ch/ (5th Sep'07).

ISL (2001). Meeting Room Projects, Interactive System Labs.
http://penance.is.cs.cmu.edu/meeting_room/ (5th Sep'07).

ISO (2003). ISO / TS 16071:2003, Ergonomics of Human-System Interaction – Guidance on Accessibility for Human-Computer Interfaces.
http://www.stcsig.org/sn/internet-ISO.shtml (5th Sep'07).

ITAW (1998). IT Accessibility and Workforce Division, Section 508 of U.S. Rehabilitation Act. http://www.section508.gov/ (5th Sep'07).

Jacobs, I., Gunderson, J., and ed., E. H. (2002). User Agent Accessibility Guidelines 1.0. Recommendation, W3C.

Jeon, J., Lavrenko, V. P., and Manmatha, R. (2003). Automatic Image Annotation and Retrieval Using Cross-media Relevance Models. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 119–126, New

York, NY, USA. ACM Press.

Johnson, B. and Shneiderman, B. (1991). Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures. In *VIS '91: Proceedings of the 2nd conference on Visualization '91*, pages 284–291, Los Alamitos, CA, USA. IEEE Computer Society Press.

Jones, D., Bench-Capon, T., and Visser, P. (1998). Methodologies for Ontology Development. In *Proceedings of IT&KNOWS Conference, 15th IFIP World Computer Congress*, pages 62–75. Chapman-Hall, London-UK.

Kalfoglou, Y. and Schorlemmer, M. (2003). Ontology Mapping: The State of the Art. *The Knowledge Engineering Review*, 18(1):1–31.

Kalyanpur, A., Hashmi, N., v, and Parsia, B. (2004). Lifecycle of a Casual Web Ontology Development Process. In *WWW Workshop on Application Design, Development and Implementation Issues in the Semantic Web*.

Kanade, T. (2007). Digital Human Modeling and Quality of Life Technology. In *Keynote address in 12th International Conference on Human-Computer Interaction*, Biejing, China.

Kaon2 (2007). An Infrastructure for Managing OWL-DL, SWRL, and F-Logic Ontologies. http://kaon2.semanticweb.org/#reasoning.

Karger, D. R., Bakshi, K., Huynh, D., Quan, D., and Sinha, V. (2003). Haystack: A Customizable General-Purpose Information Management Tool for End Users of Semistructured Data. In *CIDR*.

Karim, S., Latif, K., and Tjoa, A. M. (2007). Providing Universal Accessibility using Connecting Ontologies: A Holistic Approach. In Constantine Stephanidis, editor, *Universal Access to Applications and Services*, volume 7 of *LNCS 4556*, Biejing, China. Springer - Berlin / Heidelberg.

Karim, S. and Tjoa, A. M. (2006a). Exploiting SenseCam for Helping the Blind in Business Negotiations. In *Proceedings of ICCHP*, volume 4061) of *LNCS*, pages 1147–1154. Springer - Berlin / Heidelberg.

Karim, S. and Tjoa, A. M. (2006b). Towards the Use of Ontologies for Improving the User Interaction for People With Special Needs. In *Proceedings of ICCHP*, volume 4061 of *LNCS*, pages 77–84. Springer - Berlin / Heidelberg.

Karim, S. and Tjoa, A. M. (2007). Connecting User Interfaces and User Impairments for Semantically Optimized Information Flow in Hospital Information Systems. *Journal of Universal Computer Science: Proceedings of I-MEDIA'07 and I-SEMANTICS'07*, pages 372–379.

Keates, S. (2006). Pragmatic Research Issues Confronting HCI Practitioners When Designing for Universal Access. *Universal Access in the Information Society*, V(3):269–278.

Kelly, D., jun Yuan, X., Belkin, N. J., Murdock, V., and Croft, W. B. (2002). Features of Documents Relevant to Task- and Fact-Oriented Questions.

In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 645–647, New York, NY, USA. ACM Press.

Kieras, D. (1988). *Towards a Practical GOMS Model Methodology for User Interface Design.* North-Holland Publishing.

Klyne, G., Reynolds, F., Woodrow, C., Ohto, H., Hjelm, J., Butler, M. H., and Tran, L. (2004). Composite Capability/Preference Profiles (CC/PP): Structure and Vocabularies 1.0. Recommendation, W3C. http://www.w3.org/TR/2004/REC-CCPP-struct-vocab-20040115/.

Kutz, O., Lutz, C., Wolter, F., and Zakharyaschev, M. (2004). E-Connections of Abstract Description Systems. *Artificial Intelligence*, 156(1):1–73.

Lamping, J., Rao, R., and Pirolli, P. (1995). A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

Latif, K., Mustofa, K., and Tjoa, A. M. (2006). An Approach for a Personal Information Management System for Photos of a Lifetime by Exploiting Semantics. In *Proceedings of DEXA*, volume 4080) of *LNCS*, pages 467–477, Krakow, Poland. Springer - Berlin / Heidelberg.

Laurel, B. (1993). *The Art of Human-Computer Interface Design.* Addison Wesley, Reading, MA.

Lenzerini, M. (2002). Data Integration: A Theoretical Perspective. In *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 233–246, New York, NY, USA. ACM Press.

Limbourg, Q. and Vanderdonckt, J. (2004). Addressing the mapping problem in user interface design with usixml. In *TAMODIA '04: Proceedings of the 3rd annual conference on Task models and diagrams*, pages 155–163, New York, NY, USA. ACM Press.

Lopez, X. and Das, S. (2007). Semantic Data Integration for the Enterprise. Technical report, Oracle Corporation.

Martín González-Rodríguez, Esther Del Moral-Pérez, M. d. P. P.-R. and Pérez-Pérez, J.-R. (2003). GADEA: a Framework for the Development of User Interfaces Adapted to Human Cogni-tion Diversity. *UPGRADE The European Journal for the Informatics Professional Human-Computer Interaction: Overcoming Barriers*, IV(1):27–30.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N., and Oltramari, A. (2003). WonderWeb Deliverable D18, Ontology Library (final). Technical report, Laboratory For Applied Ontology - ISTC-CNR.

Masolo, C., Borgo, S., Gangemi, A., Oltramari, A., and Schneider, L. (2002).

WonderWeb Deliverable D17. The WonderWeb Library of Foundational Ontologies and the DOLCE ontology. Technical report, Laboratory For Applied Ontology - ISTC-CNR.

McGuinness, D. L. and van Harmelen, F. (2004). OWL Web Ontology Language Overview . Recommendation, W3C. http://www.w3.org/TR/2004/REC-owl-features-20040210/.

Menzel, C. (2003). Reference Ontologies - Application Ontologies: Either/Or or Both/And? In *KI Workshop on Reference Ontologies and Application Ontologies*.

Mindswap (2003). Photostuff - An Image Annotation Tool for the Semantic Web. http://www.mindswap.org/2003/PhotoStuff/ (5th Sep'07).

Mori, G., Paterno, F., and Santoro, C. (2002). CTTE: Support for Developing and Analyzing Task Models for Interactive System Design. *IEEE Trans. Softw. Eng.*, 28(8):797–813.

Morse, E., Michael, L., and Olsen, K. (2000). Evaluating Visualizations: Using a Taxonomic Guide. *Int. J. Hum.-Comput. Stud.*, 53(5):637–662.

Mullet, K. and Sano, D. (1995). *Designing Visual Interfaces*. SunSoft Press, Mountain View, CA.

Nicolle, C., Osman, Z., Black, K., , and Lysley, A. (2004). Learning from Internet Requirements of People with Communication Needs. In *Proceedings of the 9th International Conference on Computers Helping People with Special Needs ICCHP04, LNCS*, volume 3118, pages 121–128. Springer-Verlag.

Nielsen, J. (1994). Heuristic Evaluation. pages 25–62.

Nielsen, J. (1999). *Designing Web Usability: The Practice of Simplicity*. New Riders Publishing, Thousand Oaks, CA, USA.

NIST (2006). NIST Meeting Recognition Project. http://www.nist.gov/speech/test_beds/mr_proj/index.html (5th Sep'07).

Nokia (2007). Usability and User Experience Forum. http://www.forum.nokia.com/main/technical_services/usability/index.html (5th Sep'07).

Nonaka, I. and Takeuchi, H. (1995). *The Knowledge Creating Company*. Oxford University Press.

Norman, D. A. (2000). *The Design of Everyday Things*. London/New York: MIT Press.

Northover, S. and Wilson, M. (2004). *SWT: The Standard Widget Toolkit*, volume 1. Addison-Wesley.

Noy, N. F. and Hafner, C. D. (1997). The State of the Art in Ontology Design. *AI Magazine*, 18(3):53–74.

Obitko, M. (2007). *Translations between Ontologies in Multi-Agent Systems*. PhD thesis, Czech Technical University in Prague, Czech.

OBO (2007). The Open Biomedical Ontology Foundry. http://www.obofoundry.org/ (5th Sep'07).

Obrenovic, Z., Abascal, J., and Starcevic, D. (2007). Universal Accessibility as a Multimodal Design Issue. *Commun. ACM*, 50(5):83–88.

Obrenovic, Z. and Starcevic, D. (2004). Modeling Multimodal Human-Computer Interaction. *Computer*, 37(9):65–72.

Obrenovic, Z., Starcevic, D., and Devedzic, V. (2003). Using Ontologies in the Design of Multimodal User Interfaces. In *Human-Computer Interaction, Proceedings of IFIP Interact'03 Conference*, pages 535–542. IOS Press.

Olivé, A. (2004). On the Role of Conceptual Schemas in Information Systems Development. In *Ada-Europe*, volume 3063, pages 16–34. Springer-Verlag.

Olivé, A. (2005). Conceptual schema-centric development: A grand challenge for information systems research. In *Lecture Notes in Computer Science*, volume 3520, pages 1–15. Springer-Verlag.

Paterno, F., Mancini, C., and Meniconi, S. (1997). ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models. In *INTERACT '97: Proceedings of the IFIP TC13 Interantional Conference on Human-Computer Interaction*, pages 362–369, London, UK, UK. Chapman & Hall, Ltd.

Patrick, P. H. and Brian, M. (2004). RDF Semantics. Recommendation, W3C.

Plessers, P., Casteleyn, S., Yesilada, Y., Troyer, O. D., Stevens, R., Harper, S., and Goble, C. (2005). Accessibility: A Web Engineering Approach. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 353–362, New York, NY, USA. ACM Press.

Poulson, D. and Nicolle, C. (2004). Making the internet accessible for people with cognitive and communication impairments. *International Journal Universal Access in the Information Society*, 3(1):48–56.

Prud'hommeaux, E. and Seaborne, A. (2007). SPARQL Query Language for RDF. W3C Candidate Recommendation, W3C.

Pühretmair, F. (2004). It's Time to Make eTourism Accessible. In *Proc. of the 9th International Conference on Computers Helping People with Special Needs ICCHP'04, LNCS*, volume 3118, page 272279. Springer-Verlag.

Pühretmair, F. and Miesenberger, K. (2005). Making Sense of Accessibility in IT Design - Usable Accessibility vs. Accessible Usability. In *DEXA Workshops*, pages 861–865.

Quan, D. and Karger, D. (2004). How to Make a Semantic Web Browser. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 255–265, New York, NY, USA. ACM Press.

Rector, A., Rogers, J., P., Z., and der Haring E., V. (2003). OpenGALEN: Open Source Medical Terminology and Tools. In *AMIA Annual Symposium Proceedings 2003*, page 982. PubMed Central. PMID: 14728486.

Reed, S. L. and Lenat, D. B. (2002). Mapping Ontologies into Cyc. In *Proceedings of American Association for Artificial Intelligence*. http://www.opencyc.org/doc.

Reidsma, D., Rienks, R., and Jovanovic, N. (2005). Meeting Modelling in the Context of Multimodal Research. In *Proceedings of the MLMI'04*, volume 3361) of *LNCS*, pages 22–35. Springer Verlag.

Reynolds, D. (2007). Jena Inference API. http://jena.sourceforge.net/inference/ (5th Sep'07).

Sahoo, S. S., Bodenreider, O., Zeng, K., and Sheth, A. (2007). An Experiment in Integrating Large Biomedical Knowledge Resources With RDF: Application to Associating Genotype and Phenotype Information. In *Proceedings of WWW 2007 Workshop Health Care and Life Sciences Data Integration for the Semantic Web*. ACM Press.

Seaborne, A. (2004). RDQL - A Query Language for RDF. W3C Member Submission, W3C. http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/.

Shearer, R. (2007). Structured Ontology Format. In *OWL: Experiences and Directions 2007 Workshop, Co-located with ESWC 2007 Conference*.

Sheth, A., Agrawal, S., Lathem, J., Oldham, N., Wingate, H., Yadav, P., and Gallagher, K. (2006). Active Semantic Electronic Medical Record. In Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., and Aroyo, L., editors, *The Semantic Web - ISWC 2006*, volume 4273 of *LNCS*, pages 913–926. Springer.

Sheth, A. and Avant, D. (2003). Semantic visualization: Interfaces for exploring and exploiting ontology, knowledgebases, heterogeneous content and complex relationships. *NASA Virtual Iron Bird Workshop, CA*.

Shneiderman, B. (1996). The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *VL '96: Proceedings of the 1996 IEEE Symposium on Visual Languages*, pages 336–343, Washington, DC, USA. IEEE Computer Society.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The ICSI Meeting Recorder Dialog Act (MRDA) Corpus. In *Proc. of SIGDIAL'04 (5th SIGdial Workshop on Discourse and Dialogue)*, volume), pages 97–100.

Smith, M. K., Welty, C., and Mcguinness, D. L. (2004). Owl web ontology language guide.

SUOWG (2003). Suggested Upper Merged Ontology, IEEE: Standard Upper Ontology Working Group. http://suo.ieee.org/SUO/SUMO/ (5th Sep'07).

SUOWG (2006). Information Flow Framework Architecture, IEEE: Standard Upper Ontology Working Group.

http://suo.ieee.org/IFF/architecture.html (5th Sep'07).

Tetlow, P., Wallace, E., Oberle, D., and Knublauch, H. (2006). A Semantic Web Primer for Object-Oriented Software Developers. W3C Note, W3C. http://www.w3.org/TR/2006/NOTE-sw-oosd-primer-20060309/.

Tollis, I. and Ayache, N. (2007). Introduction to the Special Theme, The Digital Patient. *ERCIM News*, 69:85–90.

Treviranus, J., McCathieNevile, C., Jacobs, I., and ed., J. R. (2000). Authoring Tool Accessibility Guidelines 1.0. Recommendation, W3C.

UMBC (2007). Project Ontology, UMBC eBiquity: RGB Ontologies. http://ebiquity.umbc.edu/ontology/project.owl (5th Sep'07).

UMLS (1993). Unified Medical Language System. http://umlsinfo.nlm.nih.gov/ (5th Sep'07).

Uschold, M. and Grüninger, M. (1996). Ontologies: Principles, Methods, and Applications. *Knowledge Engineering Review*, 11(2):93–155.

van Kesteren ed., A. (2007). Cascading Style Sheets Object Model (CSSOM). Draft, W3C.

van Ossenbruggen, J., Troncy, R., Stamou, G., and Pan, J. Z. (2007). Image Annotation on the Semantic Web. Editor's draft, W3C.

WAI (1997). Web Accessibility Initiative. http://www.w3.org/WAI/ (5th Sep'07).

Wehrend, S. and Lewis, C. (1990). A problem-oriented classification of visualization techniques. In *VIS '90: Proceedings of the 1st conference on Visualization '90*, pages 139–143, Los Alamitos, CA, USA. IEEE Computer Society Press.

Winckler, M. A., Palanque, P., and Freitas, C. M. D. S. (2004). Tasks and scenario-based evaluation of information visualization techniques. In *TAMODIA '04: Proceedings of the 3rd annual conference on Task models and diagrams*, pages 165–172, New York, NY, USA. ACM Press.

Winston, M., Chaffin, R., and Herrmann, D. (1987). A Taxonomy of Part-Whole Relations. *Cognitive Science*, 11:417–444.

Yang, J., Zhu, X., Gross, R., Kominek, J., Pan, Y., and Waibel, A. (1999). Multimodal People ID for a Multimedia Meeting Browser. In *MULTIMEDIA '99: Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, pages 159–168, New York, NY, USA. ACM Press.

Zhou, M. X. and Feiner, S. K. (1998). Visual Task Characterization for Automated Visual Discourse Synthesis. In *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 392–399, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.