Proceedings of 10th International Conference on User Modeling, pp423-427
Edinburgh, UK, July 24-29, 2005. Springer-Verlag Berlin Heidelberg 2005

A Framework of Context-Sensitive Visualization for User-Centered Interactive Systems

Eui-Chul Jung and Keiichi Sato

The Institute of Design, Illinois Institute of Technology
350 North LaSalle Street
Chicago, Illinois 60610, USA
{jung, sato}@id.iit.edu

Abstract. This research proposes an adaptive mechanism of information visualizing that responds to context changes in knowledge-intensive work. A framework of Context-Sensitive Visualization (CSV) was introduced as a conceptual foundation for developing a middleware with three features to maximize performance of interactive systems. These features provide a mechanism for selecting appropriate content, scope, resolution, format, and timing of information delivery for effective use in changing context. In order to embed context sensitivity into the information mapping and visualization, the concept of the Context-Sensitive Object (CSO) was developed as a basic system structure for implementing the CSV.
Keywords: Context-Sensitive Visualization, Knowledge-Base, Interactive System, Context-Sensitive Object

1 Introduction

As computing and information systems become ubiquitous and pervasive in our activity space, the way users experience active delivery of functions and information embodied in the systems has become an important issue in interactive system design [1]. The performance of interactive systems is attributed to the quality of information and service delivery that responds to users' needs in changing contexts of use. Knowledge-intensive work involving complex information and decision-making particularly requires effective visual information delivery that provides appropriate selection of content, scopes, delivery timing, representation format, and information granularity. This information visualization mechanism needs to reflect users' needs that vary as context changes. However, the notion of context used in existing information systems is limited to simple states of the system or users, such as user profile, operation history, location and time.

This research introduces a conceptual framework of Context-Sensitive Visualization (CSV) as shown in Fig. 1 by incorporating the internal definition of context proposed in the authors' previous work [6]. This framework provides a structural foundation for developing context-sensitive information visualization systems with a CVS middleware embedded between the domain knowledge-base and the visualization subsystem to maximize the performance of the information system. The CSV adopts the Context-Sensitive Object (CSO) as the foundation, which incorporates users' context models in its data representation.

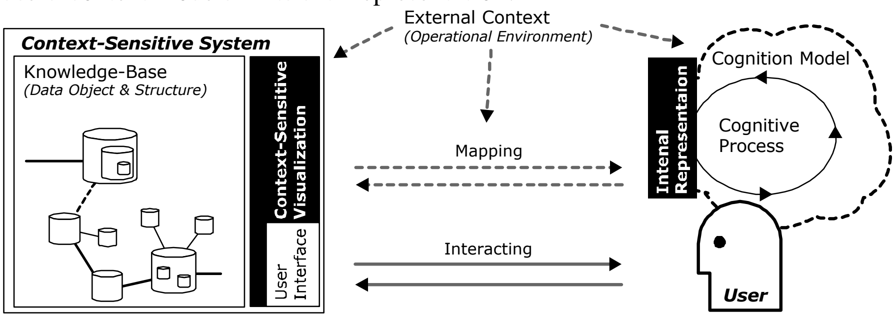

Fig. 1. Conceptual Structure of Context-Sensitive Visualization

2 Definition of Context

Diverse explanations of contexts can be found from different interests such as urban planning, usability analysis, context aware computing, and linguistics. Definitions of context found in these areas cover social, cultural and organizational aspects as well as operational aspects such as information flow, project history, and daily activity patterns. Context-aware computing includes operational environments such as location in its definition of context [2, 3]. In linguistics-based AI, contexts are considered as parameters and dynamic cues across sentences for contextual reasoning [4]. In theses examples, contexts are considered as sources of influences that affect the performance of systems. Such explanations can be categorized as an external definition since it primarily refers to external factors of human cognition. These externally defined contexts are only meaningful to a human or machine agent when they are recognized and associated with the current cognitive state and actions of the agent. Internal definition of context, therefore, considers context as a set of mental models within the human cognition system activated by the recognition of the current situation [5, 6]. For example, a person's selection of transportation to the airport depends on several aspects of context such as mental models of causal relations among possible events, geographic relations of routes and traffic conditions, and cost and convenience.

3 Context-Sensitivity in Interactive Systems

When the information is visualized in coherence with contexts or user's mental models evoked by the situation, the content of the information is effectively transferred to the user. The Context-sensitive Object (CSO) as depicted in Fig. 2 was introduced as the basic structure of the CSV.

The CSO consists of the user object and contextualized knowledge frames with the operations that bridge the knowledge-base and visualization engine. Knowledge-base contains knowledge frames and their meta-frames [8]. The CSO is activated by trigger elements such as goals, tasks, state changes, and actions from external sources. Depending on the goal of an operation, relevant knowledge frames in the domain knowledge-base are selected by the CSO operation for delivery to the user. User object built in the CS Middleware are collections of meta-models and models of contexts. The CSO operation then selects or activates a set of context models based on the selected domain knowledge and the external information. The selected knowledge elements and external information are then mapped on and positioned in the activated context models to develop contextualized knowledge as internal representation in the CSO. The internal knowledge representation and the associated context models are then mapped onto the visual objects and structure by the operation of the visualization engine. Methods for context model representation and the mechanism of the four operations defined in the diagram are critical to develop the CSO. Frames for context models must be generated dynamically for diverse situations based on consistent parametric structures stored in meta-context model. Parameters are detachable variables that store data for contexts, conditions, and states.

The benefits of introducing the CSO are: 1) keeping information mapping consistent with human cognitive models to enhance the effectiveness of the information delivery, and 2) representing contexts explicitly through the visualization process to allow a user's interaction with deeper levels of the visualization mechanism.

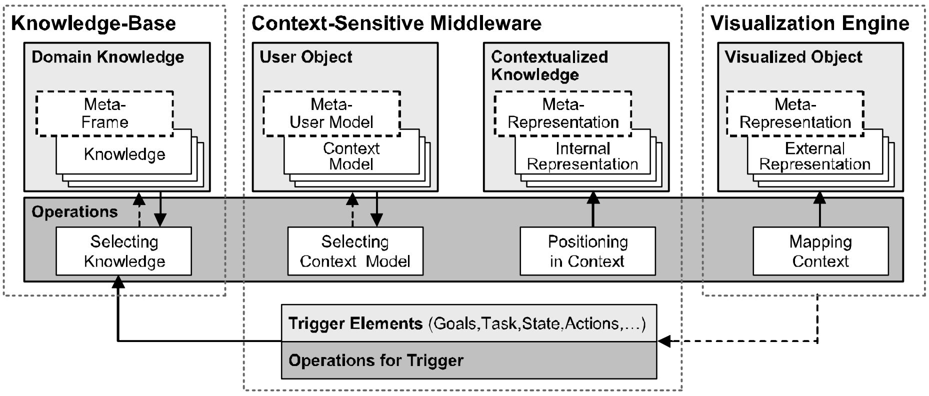

Fig. 2. Basic Structure and Information Flow of Context-Sensitive Object

For implementing the operations, contextual reasoning can be introduced as one of the methods for selecting contexts and positioning domain-knowledge and information in the selected context. Linguistic based AI defines three general reasoning mechanisms: Localized Reasoning, Push & Pop, and Shifting [4]. By controlling contexts as parameters dynamically, these mechanisms can be incorporated into the operations for generating contextualized knowledge. *Localized Reasoning* contains basic mechanisms how systems control meta-frames and meta-user models to answer for triggering elements internally. Calculating the optimized route to destination for user's goal is an example of *localized reasoning*. However, sometimes a user wants to manipulate contexts directly to get more meaningful knowledge by adding and/or removing contextual parameters. *Push & Pop* works here. For instance, if a driver finds the routed road is suddenly not available, a driver must control the state of road (the value of meta-frame) directly to get another route. A context of road availability is *pushed* by the user. Or, a context can be *popped* out in some cases. *Shifting* provides users with new viewpoints of the knowledge by controlling the value of meta-frame and meta-user model for context data. If a user is driving on a busy road, time-based distance representation is the optimal visualization solution. However, an indexical representation for gas station is better for visualization, if the fuel gauge is approaching empty.

4 System Architecture for Context-Sensitive Visualization

The basic system architecture in Fig. 3 was developed to implement a software platform for simulating and evaluating the CSV concept and functional subsystems. The CSV-based system can be effectively implemented for diverse applications for knowledge-intensive work such as business, engineering, education, communication, and medical work where contexts take critical roles. As shown in Fig. 3, it consists of three parts: Visualization Engine, Data-Processing Engine, and DBMS. Visualization engine has the component of stage creator, interaction controller, and manager for visual objects, attributes, and spaces in visual representation. Users can interact with the displayed information objects to make them consistent with their mental models to enhance their performance.

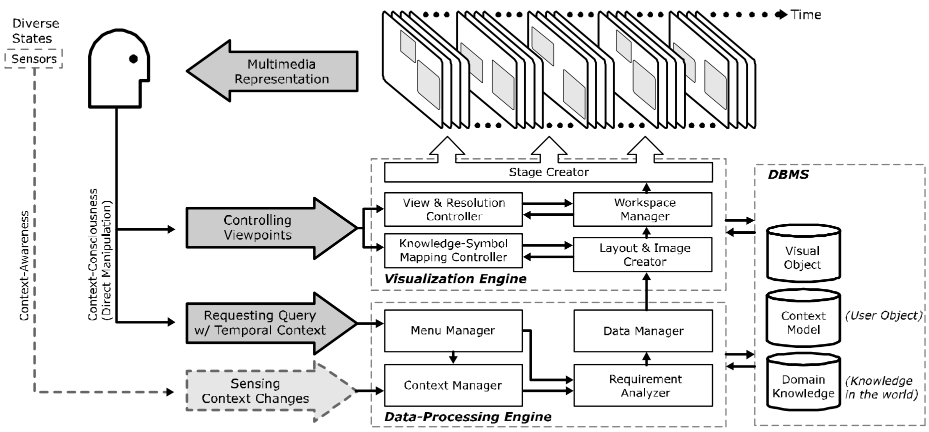

Fig. 3. System Architecture for CVS-based information systems

Data-processing engine is a collection of functional modules such as managers for menu and data, and analyzer for interpreting information. This part is for analyzing the external situations through sensors, actions, and internal contexts like gathered patterns of history to decide the appropriate context models for visualization in the situation. A context manager handles these processes and an interface manager enables a user to control the initial index of context. A data manager works as a gate between DBMS, visualization engine, external DBMS, and requirement analyzer.

DBMS consists of knowledge-base encoded in the CSV, context model, and visual object. It also has temporal DBMS for storing the trigger elements and history of users' action. The CSV middleware provides the control of selecting: 1) knowledge content, resolution, format, and timing for effective information delivery in changing context, 2) interaction methods to control and monitor information for capturing context for understanding information and making better decision by enhancing users' cognitive capacity, and 3) data accumulation methods to record the history of users' interactions for elicitation of their mental models through the evolution of context models.

5 Conclusion and Perspectives

This research introduced a conceptual framework for developing context-sensitive visualization systems. In the course of the framework integration, internal definition of context was explained as a set of mental models and incorporated as the context representation method in CSV. In order to develop the CSV middleware applicable to diverse interactive systems, further studies such as building the knowledge class libraries for context models, and the mapping and visualization logics between data and visual objects with case studies must be conducted.

References

1. Dourish, P.: Where the Action is; The Foundations of Embodied Interaction. MIT Press, Cambridge, MA (2001) 1-23
2. Schmidt, A., Beigl, M., and Gellersen, H. W.: There is more to Context than Location. Proceedings Workshop on Interactive Applications of Mobile Computing (IMC' 98), Rostock, Germany (1998)
3. Benerecetti, M., Bouquet, P., and Ghidini, C.: Contextual Reasoning Distilled, Journal of Theoretical and Experimental Artificial Intelligence, 12(3) (2000) 279-305
4. Kokinov, B.: Dynamics and Automaticity of Context; A Cognitive Modeling Approach. In Lecture Notes in Artificial Intelligent, Springer Verlag (1999)
5. Sato, K.: Context-Sensitive Design; Bridging Viewpoints for Human-Centered Design, Proceedings of the FutureGround 2004 Conference, Design Research Society, Melbourne (2004)
6. Kobsa, A.: Generic User Modeling Systems, User Modeling and User-Adapted Interaction 11 (2001) 49-63
7. Bresillon, P.: Making Context Explicit in Communicating Objects, in Communicating with Smart Objects: Developing Technology for Usable Pervasive Computing Systems, Kogan Page, London (2003)
8. Kashyap, V. and Sheth, A.: Semantic and Schematic Similarities between Database Object: A Context-Based Approach, The VLDB Journal, Springer-Verlag (1996) 5:276-304