A FRAMEWORK FOR INTELLIGENT CHANGE MANAGEMENT IN THE CONSTRUCTION PROCESS

Q. Y. Cai[1], F. F. Ng, H. Chen and Z. K. Ding

Department of Real Estate & Construction, The University of Hong Kong, Pokfulam Road, Hong Kong

During a project lifecycle, changes are inevitable due to various uncertainties, complexity of tasks and mobility of people involved. For effective change management, project stakeholders must know the latest changes, where the changes occur and what the impacts are in terms of time and cost. However, current internet-based project management systems always lack the flexibility to handle dynamic changes and collaborative processes. A typical scenario in handling changes is to inform all the stakeholders by broadcasting messages. This may incur two potential problems. First, the messages may be neglected due to information overload. Second, no implications of the change are suggested in the messages. This paper proposes a framework for effectively managing changes in the construction process, where ontology serves as the theory core layer, intelligent agents are in the application layer and customized workspace is presented as an interface layer. Using applications developed from this framework, users can get customer-tailored change reports and recommendations to cope with the changes.

Keywords: change management, construction process, intelligent agent, Internet-based project management system, ontology.

INTRODUCTION

During a project lifecycle, a huge amount of information is produced. All these information are highly interdependent. A change initiated in certain stage always affects the successor in following stages. For example, the change of a door in the architectural design may cause the acoustic designer to change the design for noise control, or cause the contractor to purchase a new product from the suppliers. What we concern here is the information changes instead of organizational changes. From organization's perspective, change management is about how the organization changes itself (may cover the people, enterprise or the business processes) to adapt to the environment. One example of this change management is the proposition of Business Process Reengineering (BPR), which desires to achieve higher productivity in organizations through re-structuring the business processes. In our research, change management is about how the stakeholders manage the changes of project information during the construction process.

In a project, if every piece of information is viewed independently, then it is a new message for every instance. However, as mentioned above, project information is interdependent. Hence, a message about "change" contains certain implicit elements, i.e. its relationships with previous information and current or future information. In general terms, "change" should indicate the transition of information from "previous one" to "current or future one". For example, a simple "change" may be the change of

[1] kikicqy@hkusua.hku.hku

Cai, Q Y, Ng, F F, Chen, H and Ding, Z K (2004) A framework for intelligent change management in the construction process. *In:* Khosrowshahi, F (Ed.), *20th Annual ARCOM Conference*, 1-3 September 2004, Heriot Watt University. Association of Researchers in Construction Management, Vol. 2, 995-1003.

state (say location) of a partition wall. To evaluate the impact of such "change", then we should compare the changed state with the previous state. For example, the impact of the change would involve the change of size of rooms or spaces bounded by that wall.

So, change management is not just about information flow. We need to be able to track the changes over time. For effectiveness, we should be able to evaluate the impact of each change comparing to its previous states. In order to deduce the impact, we need to understand the relationships between various elements and their construction processes.

The problem in current Internet-based project management system is they treat changes as new coming messages and broadcast them to all the stakeholders. This may incur two potential problems: first, the changes are neglected due to the information overload; second, no implications are suggested. Therefore, users may need excessive time to process the information: to filter the information to fit for personal needs, to evaluate the impacts of changes on the construction cost and time, to identify who they should inform of and get authorization, and to negotiate with others in case of conflicts. All of these always become the stimulus for possible delay, extra costs and even disputes in a project.

This paper proposes a framework for intelligent change management in the construction process, which is supposed to benefit stakeholders in three aspects: identification, evaluation and authorisation (Atkin, Gravett and Smith 1998). By this means, users can get customized change reports, which state the latest changes, where the changes occur, whom they will affect, and what their impacts on the construction time and cost. By the term of “intelligent”, we mean stakeholders can get recommendations from the system. Therefore, the system is proactive, rather than reactive, to the changes and their impacts in the construction process.

In the following parts, first, we illustrate the rationale of ontology, intelligent agents and customized interface, and also state how they can be used to support intelligent change management. Then we propose the framework, followed by illustration of the components. We discuss the related works and finally draw the conclusion.

ONTOLOGY

In the framework, ontology is the backbone to support the exchange of information from different perspectives. Why can ontology support this, what is it, and how it can be developed? This section talks about these in more details as follows.

The Need for Ontology

Ontology is a term borrowed from philosophy, where it is referred to what the world exists and how it is configured. It gained popularity in the Artificial Intelligence (AI) domain in the late 1990s when people found the knowledge-based system was not reusable. This is because different knowledge is developed in its own domain and point-to-point translations between pairs of applications can’t facilitate real knowledge sharing (Ciociou, Gruninger and Nau 2000). What is needed is to find the basic concepts/meanings served as building blocks for knowledge-base systems so as to enable knowledge sharing in an unambiguous way. When agents (human or artificial) all agree on these basic meanings/semantics, they can communicate with each other without any misunderstanding or semantic clashes (Fonseca 2001).

The result of such an explicit formalization of our mental models is what the Artificial Intelligence (AI) community calls ontology (with a lower-case o). A widely accepted definition of ontology is stated by Gruber (1993, cited in Fensel 2003), “An ontology is a formal, explicit specification of a shared conceptualization.” Here a "conceptualisation" refers to "an abstract model of some phenomenon in the world which identifies the concept related to the phenomenon." For example, when we say an "elephant", in our minds, we can picture what an elephant looks like in the real world, and we can make sure that we refer to the same kind of animal. By this means, people from different perspectives can reach consensus through commitment to that ontology.

What An Ontology Looks Like

Understanding the ontology is a common vocabulary to describe the information content, you may wonder what it looks like. There is no standard form to describe an ontology, but from the definition we know it necessarily contains the vocabulary in a domain – terms, synonyms, antonyms, and the relationships between them. A typical way to structure ontologies is to organize them as hierarchies, with objects represented as classes, the features as slots, and the relationships between classes as facets/constrains. For more references, please refer to Kayser (1998), Sowa (2000) and Shum (2001).

How To Develop Ontologies To Support Change Management

In order to develop ontologies to support change management in the construction process, this research uses two methods, one is to borrow ideas from Industrial Foundation Classes (IFC)[2] 2x, and another is to integrate the domain ontology and task ontology to develop application ontology. IFC 2x has defined some terms regarding the process model, e.g. IfcProcess, IfcTask, IfcChangeOrder, etc. However, it only models physical activities. To model the process and changes in the information level, we may consider making use of some available task ontologies, and matching them to the AEC industry domain. Such task ontologies have been developed in other industries and we don't have to start from scratch. For example, Rajpathak (2001) has developed the scheduling task ontology, and it is domain and method independent. By integrating them into the AEC industry, we can use these task ontologies to model the changes and their impacts in terms of time and cost in the construction process.

Supporting tools to develop ontologies can be Protégé 2000[3], Ontolingua, WebOnto, OntoEdit, etc. In our research, we use Protégé 2000, which is a graphical tool for ontology editing and knowledge acquisition with new and evolving Semantic Web language, i.e. OWL and ezOWL plugin. Comparing with other software, Protégé 2000 concentrates on the concept models instead of the syntax to be used on the web. For agents to access in and exchange information, these ontologies should be written in sort of languages, e.g. OWL4 (Web Ontology Language), RDF5 (Resource Description Framework), XML6 (Extensible Markup Language), etc. In Protégé 2000, the developed ontologies can be exported as OWL file by simply selecting a "save as…" item from a menu.

INTELLIGENT AGENT

What Is An Intelligent Agent

An intelligent agent is defined as "an autonomous, (preferably) intelligent, collaborative, adaptive computational entity. Here, intelligence is the ability to infer

---

[2] http://www.iai-ev.de/spezifikation/IFC2x/index.htm
[3] http://protege.stanford.edu/index.html
[4] http://www.w3.org/2001/sw/WebOnt/
[5] http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/#intro
[6] http://www.w3.org/TR/2004/REC-xml-20040204/

and execute needed actions, and seek and incorporate relevant information, given certain goals."[7] As stated in the definition, the key features of intelligent agents include:

- **Autonomy:** An important element of the autonomy feature is the pro-activeness. i.e., their ability to take the initiative rather than acting simply in response to their environments. Agent can formulate its own goals and act to meet the client's requirements.
- **Co-operation**: Instead of having a single agent to carry out the task, you can have multiple agents to co-ordinate with each other to fulfil it.
- **Learning**: Agent can learn as they react/interact with external environment.

Having above features, agents are widely used in workflow management, air traffic control, business process re-engineering, information retrieval and management, e-commerce, personal digital assistants, e-mail filtering, command and control, smart databases and scheduling/diary management.

Agent Architecture In Our Research

In our research, there is heterogeneous agents co-ordinating with each other to fulfil different objectives. We adapt the architecture of RETSINA multi-agent system (MAS) developed by the Intelligent Software Agents Lab at Carnegie Mellon University's Robotics Institute[7].The architecture is shown as below:

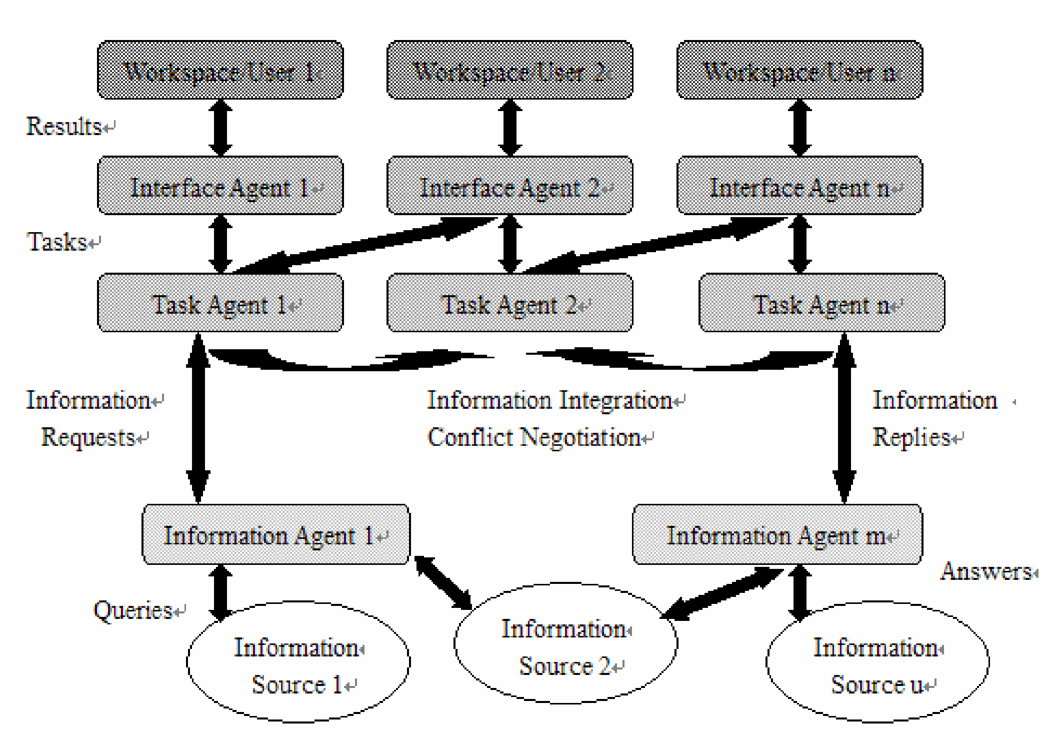

Figure 1: Proposed architecture of heterogeneous agents for intelligent change management

In this architecture, interface agents are responsible for interacting with users and task agents. They can observe the working style of users, and display the results as the users want. Task agents represent the users to execute some tasks, e.g. notifying the stakeholder for some new changes, negotiating with other parties in case of conflicts, etc. They may be mobile agents, which migrate from one computer to another. Information agents extract information from the information sources, and respond to the task agents in terms of requests. These software agents operate on some standard

[7] http://www-2.cs.cmu.edu/~softagents/intro.htm

platforms, such as XML (Extensible Markup Language), CORBA (Common Object Request Broker Architecture), Java, etc, and they communicate with each other in some common languages, e.g. KQML (Knowledge Query and Manipulation Language), ACL (Agent Communication Language), IIOP (Internet Inter-ORB Protocol), XML-based scripting language, etc.

CUSTOMIZED INTERFACE

Due to the fact that different users may carry out different tasks, and have different requirements in different domains of interest, existing design methods do not necessarily support designing such User Interfaces (UIs). Traditional UIs always display the information regardless of user preferences, cognitive style, languages, culture, habits, and system experience. Universal design tries to overcome these problems by considering multiple parameters. However, this multiplicity of parameters dramatically increases the complexity of design works, and universal design doesn't identify and manipulate these parameters in a structured way. Some researchers (Furtado, et al 2001) propose the method to integrate ontologies into universal design to support the different presentation style, UI structures, etc.
In our research, we are going to use interface agents as the facilitators of the customized workspace, where ontology is the backbone to support it. After agents detecting some changes, they may want to deliver them to the users in an effective way. Then a customized Web page is ideal for presenting the changes, where every stakeholder can get his/her customer-tailored services.

FRAMEWORK FOR INTELLIGENT CHANGE MAN AGEMENT

In order to achieve above goal, a nutshell that indicates the rationale behind the framework is proposed as figure 2. We argue that in future Internet-based project management systems, different stakeholders will have their own agents on behalf of them to exchange information and update the changes. The agent will have the ontology as its backbone, and customized workspace as its representation interface. We illustrate this idea as below:

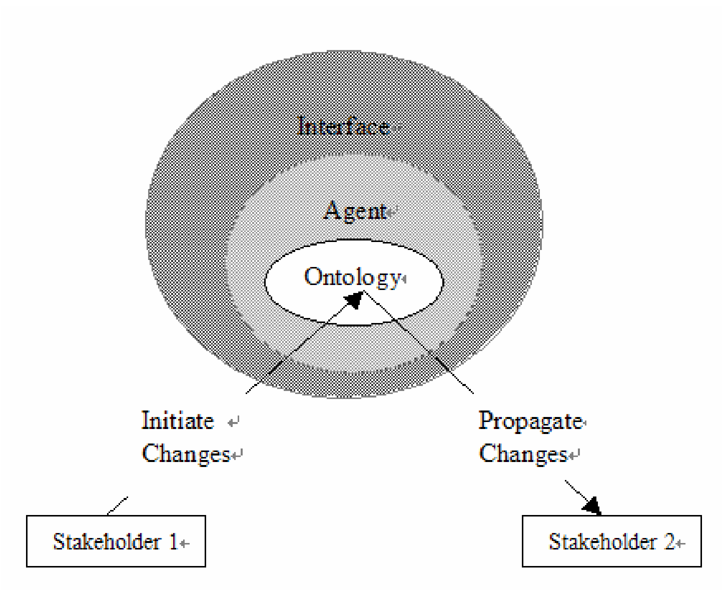

Figure 2: A nutshell to support intelligent change management from different

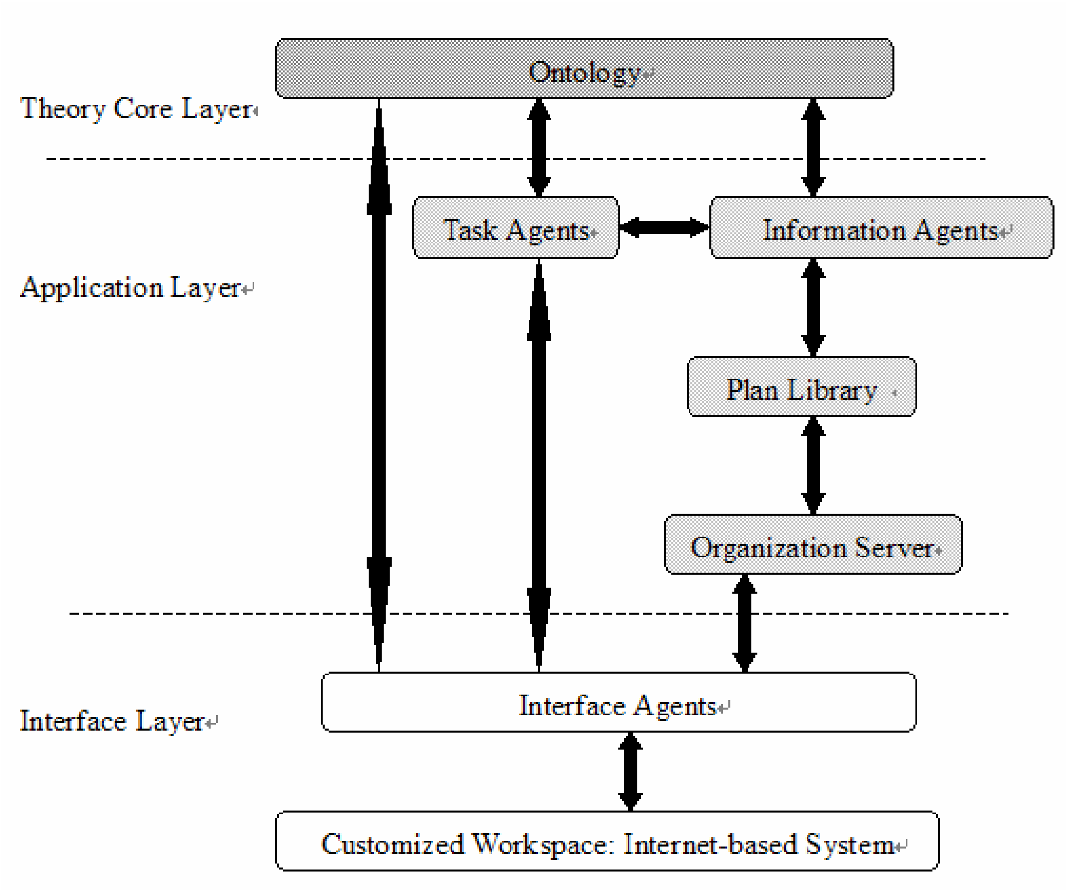

Figure 3: Proposed framework for intelligent change management

As figure 3 shows, the framework is consisted of three layers, i.e. interface layer, application layer and theory core layer. We discuss them in more details:
Ontology (Theory core layer)
This layer is the backbone for other layers. It defines a common vocabulary for different agents to communicate with each other, and supports the users to customize their interfaces based on their preferences. In this layer, terms are defined and used consistently in naming objects, e.g. task, pre-condition, post-condition, etc. This is the requisite for an explicit representation of a shared understanding of different domains.
Information Agents (Application layer)
During the construction process, the Information Agents create/collect ontologies for their masters from the project resources, i.e. documents, drawings, mail, etc. They group the planed process into the Plan Library, and query/answer the particular requests from the Task Agents. This process can be automatically or semi-automatically with the support of current/developing technologies.
Plan Library (Application layer)
This is the library of the workflow engine, where the process templates are collected and stored by the Information Agents. In this library, tasks are decomposed according to the planed process. There are some constraints in order to execute the tasks, i.e., preceding tasks must be completed, the pre-conditions must be fulfilled and the post-conditions will be achieved upon task completion. Moreover, the responsible parties, the time/schedule and the cost that associate with the tasks are presented in this library. When there are new changes occur, this library has all the planed time, cost and process templates for Information Agents to extract the related information.
Organization Sever (Application layer)

This server allows the Information Agents to collect information across the organization boundaries, and stores the plan library specific to that organization, where the Interface Agents can be designed to fit for specific purpose.

Task Agents (Application layer)

If we classify the Information Agents as internal agents for specific organizations to structure information for them, then the Task Agents are external ones for different organizations to exchange information and negotiate with each other in case of conflicts. Because changes are initiated by one organization, and then propagated to other ones, we need this kind of agents to roam from one organization to another to collect the updated information, and report the changes to stakeholders.

Interface Agents (Interface layer)

Interface Agents represent the users to respond to the Task Agents, and facilitate the results to be displayed in a way that satisfies the users' preferences. For example, users can choose the preference as "manual input" or "selection" to display the options.

Customized Workspace: Internet-based System (Interface layer)

After the manipulation of Interface Agents, different users can have their own customized Web pages, and this is integrated into the Internet-based project management systems.

With the implementation of the framework, changes are supposed to be delivered to the right hand at the right time, thus possible delay due to information neglect can be avoided, fast reaction can be initiated, and efficient authorization process can be enabled.

DISCUSSION AND RELATED WORKS

Some researchers have realized the problems in current project management systems, and tried to make explicit the business processes for the sake of better managing the changes.

Traditionally, most business processes have been modeled only in terms of building components (CAD drawings) and physical activities (project schedule). IFC, aecXML can be regarded as such research efforts. Some researchers (Barrón and Fischer 2001) argue that business processes should also include management activities dealing with documents such as RFIs, submittals, change orders and billings. Therefore people will not only know "what to do", but the impact of physical activities on project schedule and costs can be well understood, controlled or predicted. "These models will then enable project stakeholders to prioritize their time and efforts by providing them visibility into the impact their management activities will have on physical activities." (p. 7) Such research decomposed the business process into various levels ranging from the project phases to a specific transaction at the document level, and then to an activity at the information level.

After making explicit the business process, in order to reflect the impact of changes, some researchers use the Excel spreadsheet to automatically update the affected data, e.g. Barrón and Fischer (2001). They use tables as the schema to capture the implications of changes. For example, the column may state the business process, and the rows may state resource, actor, cost, duration time, output documents, etc. Different tables can be linked together to interpret the implications. It is a very powerful way to reflect the impact of changes.

However, is there any theory behind the use of spreadsheet to model the process and manage the changes? Is there any better way for information change management?

We argue that what the spreadsheet represents is a preliminary ontology. The possible theory behind it may be ontologies. Based on this, our research tries to develop a prototype for information change management, which integrates the ontology, intelligent agents and customized interface.

CONCLUSION

Due to the fact that various uncertainties, complexity of task and mobility of people involved in the construction process, changes are inevitable. In this paper, we try to illustrate how a framework consisted of three layers can enable the immediate identification of changes, and give customized recommendations of evaluation and authorization for project stakeholders. Although at current stage, the research result is quite limited, we state the feasibility of the framework by illustrating the rationale, and the available technologies to implement it.

REFERENCES

Atkin, B, Gravett, J and Smith, D (1998) Benchmarking Best Practice: Cost and Change Management. Salford UK: Construct IT Centre of Excellence.

Barrón, A and Fischer, M (2001) Potential Benefits of Internet-Based Project Control Systems – A Study On Change Order Processing, CIFE Technical Report #126, Stanford University, http://www.stanford.edu/group/CIFE/Publications/index.html [April, 2004]

Ciocoiu, M, Gruninger, M and Nau, D S (2000) Ontology for Integrating Engineering, Journal of Computing and Information Science in Engineering, 1(1), 12-22.

Fensel, D (2003) Semantic Web Services: A Communication Infrastructure for eWork and eCommerce, In: Lovelle, J M C et al (Eds.), Web Engineering: International Conference, ICWE 2003, July 14-18, 2003, Oviedo, Spain, Springer, 1-7.

Fonseca, F T (2001) Ontology-driven Geographic Information Systems, Unpublished PhD Thesis, University of Maine.

Furtado, E, Furtado, J. J. V, Silva, W. B, et al. (2001) An Ontology-Based Method for Universal Design of User Interfaces, Workshop on Multiple User Interfaces over the Internet: Engineering and Applications Trends,

http://www.cs.concordia.ca/~seffah/ihm2001/papers/ [May, 2004]

Kayser, D (1998) Ontologically Yours, In: Mugnier, M L and Chein, M (Eds.), Conceptual Structures: Theory, Tools and Applications, 6th International Conference on Conceptual Structures, ICCS'98, Montpellier, France, 35-50

Rajpathak, D, Motta, E, Zdrahal, Z and Roy, R (2003) A Generic Library of Problem Solving Methods for Scheduling Applications, In: the Second International Conference on Knowledge Capture, KCAP'2003, Oct. 23.25, 2003, Florida, USA, 113-120, http://kmi.open.ac.uk/people/dnyanesh/ [April, 2004]

Shum, S B, Domingue, J and Motta, E (2001) Ontology-driven Knowledge Management: Philosophical, Modelling and Organizational Issues, In: Roy, R (Eds.), Industrial Knowledge Management: A Micro-level Approach, London, New York: Springer, 265-280.

Sowa, J F (2000) Knowledge Representation: Logical, Philosophical, and Computational Foundations. Pacific Grove, Calif.: Brooks/Cole.