Dynamic Management of e-Business Services: A Situation Calculus Based Approach

Bing Li, Junichi Iijima
Department of Industrial Engineering and Management, Graduate School of Decision Science and Technology, Tokyo Institute of Technology, Japan {li.b, iijima.j}@m.titech.ac.jp

ABSTRACT
In the era of e-Business, the traditional business services are greatly challenged by the ever-increasing demands from customers with various backgrounds and personalities. Large numbers of new e-Businesses are driven by the needs of customers. This paper proposes a Situation Calculus based approach to dynamically manage e-Business services in the ubiquitous and networked environment. By employing this formalism, the approach can implement service model verification and automatic composition.

Keywords: dynamic management, e-Business service, Situation Calculus

1. INTRODUCTION
The global economies are gradually evolving to the era of information. The traditional business models are necessarily substituted or complemented by e-Business services with the development of the ubiquitous and networked environment.

In this new economy, there are some existing or emerging challenges. One of them is the degree of customer-orientation. Large numbers of customers are accessible to the Internet and they demand kinds of services to satisfy their various needs. Customer-oriented services are required from the perspective of demands. On the other hand, with the heated competition among service suppliers, customer-orientation will also be driven to a higher degree when the suppliers are seeking ways to differentiate their services. Higher degree of customer-orientation will greatly necessitate the various personalized services.

With the increasing amounts of e-Business services, dynamic management of these services will be necessary so that the continuously changing business needs can be satisfied and the customized services can be fast provided. This paper provides an approach to implement the dynamic management of e-Business services. The approach is based on Situation Calculus to enable some degree of automation and business intelligence, and thus to implement the dynamic management.

This paper is organized into five sections as follows. Section 2 provides a motivating scenario to understand the necessity of dynamic management of e-Business services. Section 3 defines what dynamic management means and explains our approach in detail. Section 4 discusses some related work and Section 5 gives the conclusion and future work.
so on. That is to say, the company of e-Mover will build its own service by integrating many existing services provided by other companies or agencies.

Generally in the e-Business environment, the customer’s needs will not be fully satisfied by an isolated service. The collaboration and integration of the existing services are greatly necessary in order to reach a higher degree of customer satisfaction. Moreover, with increasing degree of differentiation of services, the services within a company will not be sufficient, and it is important to seek help or collaboration from services provided by other companies. All these above will be a problem about dynamic management of e-Business services.

3. DYNAMIC MANAGEMENT

3.1 Concept and Content

In this paper, the concept of dynamic management is aiming to implement a higher degree of customer satisfaction by providing customer-oriented services in e-Businesses. The content of dynamic management will mostly include automatic service composition and model verification. It can also be extended to include service discovery, monitoring and control in the future work.

The motivation scenario in Section 2 covers the automatic composition of services. This is to compose services automatically from the existing isolated services inside or outside an enterprise. In order to enable this automatic composition, formal descriptions of services are prerequisite. A formal description refers to specifying services by employing mathematical logics. With such a formal specification, services are described precisely and unambiguously. Logical reasoning can be implemented, which enables the automatic composition.

Service model verification is to check if a service model can satisfy the service specification. That is, given the initial situation of a service and the formal service specification, see if the goal situation can be satisfied. The model verification can be applied at two levels: one is the lower level inside a service, i.e., to check if the constituent activities can work to reach the goal of a single service; the other is the upper level across services, i.e., to check if the constituent services of a composite service can collaborate to reach the overall goal. This paper is focused on the latter one with a greater granularity. That is, a service is abstracted as a black box with formal description about its interface with other services.

3.2 Approach

In this paper we propose a Situation Calculus based approach to implement the dynamic management of e-Business services. E-Business services will be formally specified in Situation Calculus. Based on formal descriptions, some reasoning work can be performed automatically and thus enables the dynamic management of services.

Situation Calculus is usually considered as a dialect of First Order Language although it also has some second order features. This formalism is widely used in AI related fields and provides a modeling method for dynamical domains. The core concepts include actions, situations and fluents. Actions are represented by action functions, and specified by precondition and successor effect axioms; situations are world histories represented by the sequence of actions; fluents are functions and predicates that vary with the situation [1].

Most of the domain model is to specify actions with precondition and successor state axioms, which usually consist of fluents representing the world states. The reasoning mechanism is to interconnect the actions by successor states of one action with the precondition of another. Thus, given an initial situation of a domain world and the formal specification about actions, the successor states of executing some actions can be inferred; given the initial situation and the goal situation, possible execution of actions can also be obtained. The former case is model verification while the latter is automatic composition. Thus the dynamic management explained in Section 3.1 can be implemented by employing Situation Calculus.

Strength of this formalism is reasoning about actions. The domain world is specified mostly about the involved actions, and execution of these actions will result in the changes of the world states. In the ubiquitous and networked e-Business environment, services can be represented with actions in Situation Calculus. They are formally described with precondition and successor state specification. With the formal specification about e-Business services, some intelligent functions can be enabled, which also include the dynamic management.

3.2.1 Solution of Motivating Scenario

To solve the problem in the motivating scenario by this approach, the prerequisite is the formal description of the e-Business services in the domain. In this example, there are several related search services:

Search service related to the house category, which can be provided by a rental company. This service is denoted as serv_house;

Search service for the environment category, which can be provided by some related companies or facilities. This service is denoted as serv_env.

For simplicity, we will only show how to specify these two services by using Situation Calculus in Prolog programming. The traffic related service could also be specified this way. Furthermore, in the domain with much more existing services, those services can similarly specified in this logical syntax.
For the service \textit{serv\_house}, represented as 
\[ \text{serv\_house}(Hreq), \]
Precondition of \textit{serv\_house} is,
\[ \text{poss}(\text{serv\_house}(Hreq), S) :- \text{validHR}(Hreq, S). \]
Successor State of \textit{serv\_house} is
\[ \text{satisfyHR}(\text{Result}, \text{Hreq}, \text{do}(A, S)) :- A=\text{serv\_house}(\text{Hreq}); \text{satisfyHR}(\text{Result}, \text{Hreq}, S). \]

For the service \textit{serv\_env}, represented as 
\[ \text{serv\_env}(Ereq), \]
Precondition of \textit{serv\_env} is,
\[ \text{poss}(\text{serv\_env}(Ereq), S) :- \text{validER}(Ereq, S). \]
Successor State of \textit{serv\_env} is
\[ \text{satisfyER}(\text{Result}, \text{Ereq}, \text{do}(A, S)) :- A=\text{serv\_env}(\text{Ereq}); \text{satisfyER}(\text{Result}, \text{Ereq}, S). \]

\textit{Hreq} is the user’s requirements for the house; \textit{Ereq} is the user’s requirements for the environments; \textit{Result} is the output from the search services; \textit{poss} is a predefined binary predicate representing that an action is possible to execute at a certain situation; \textit{satisfyHR} or \textit{satisfyER} is the predicate stating that \textit{Result} meets the requirements \textit{Hreq} or \textit{Ereq}; \textit{validHR} or \textit{validER} is to assert that \textit{Hreq} or \textit{Ereq} is valid requirement for the system to process.

For the integrated search service provided by the e-Mover, it can be represented as 
\[ \text{serve}(\text{UserId}, \text{Hreq}, \text{Ereq}), \]
\textit{UserId} is the user account for a specific user, which can be mapped to the user profile that has been registered in the system and can be used to prioritize the search results and also provide potential recommendation.

The initial situation to serve the user Tom with id of \textit{tom2007} can be represented as
\[ \text{validUID}(\text{tom2007}, s0). \]
\[ \text{validHR}(\text{hreq}, s0). \]
\[ \text{validER}(\text{ereq}, s0). \]

The goal situation of this service will be
\[ \text{satisfyHR}(\text{Result}, \text{hreq}, S), \text{satisfyER}(\text{Result}, \text{ereq}, S). \]

Then by reasoning in Situation Calculus, the whole service should include the services of \textit{serv\_house} and \textit{serv\_env} to reach the final goal. This illustration simplifies the problem in order to give a brief explanation. When the number of available e-Business services greatly increases, only depending on human processing cannot satisfy the expanding customer’s needs. Thus it will greatly necessary to dynamically manage these services in some degree of automation. Service automatic composition and model verification will enable the provision of customer-oriented and satisfactory services.

3.2.2 Hybrid Implementation

The approach can be implemented in a prototype system that has a Java Interface and Prolog Reasoning Background. The architecture on this dynamic management system is illustrated in Fig.1. The Java interface provides the entry for service model verification (MV) and automatic composition (AC). To perform these functions, the system will query the underlying prolog engine in ECLiPSe. The Situation Calculus based formal descriptions of the e-Business
services will form the knowledge base for Prolog reasoning.

4. RELATED WORK

There are some existing research fields related to the e-Business Management. The approach in this paper is greatly motivated by some advancement in Business Process Management (BPM) [2]. We are also doing the research in verifying business processes by employing Situation Calculus [3]. In the e-Business environment, a service model is a process model and the constituent activities in a process can be services. Due to these connections, management of e-Business services can share some concepts from the BPM related research and practice [4, 5, 6].

Situation Calculus is employed for its formal foundation and strength of reasoning about actions. Formal specification will remove the space of unambiguity and enable the potential analysis [7, 8]. In our approach, just such formal descriptions about services make it possible to dynamically manage the existing services. In the domain world modeled by Situation Calculus, actions are specified in a compact form and reasoned about its precondition and successor states. E-Business services can be represented as actions and thus enable model verification and automatic composition in dynamic management. Furthermore, some former researches can provide some pointers to the application of Situation Calculus in the business world [7, 9, 10].

Semantic Web is also necessary to mention. The related researches are trying to add or mine the semantic information from the web [11, 12]. In some extent, e-Business services can possibly be described in OWL/RDF. With the future development, it is also possible to extend our approach to transform from an OWL formatted service description to a Situation Calculus based one. Thus this approach is also promising in a full-fledged environment for semantic computing.

5. CONCLUSION AND FUTURE WORK

In this paper we propose an approach for dynamic management of e-Business services in the ubiquitous and networked environment. This approach employs Situation Calculus to formally describe services. Based on such a formal specification, the automatic composition and model verification can be performed. In this way, even with the exponential increase of e-Business services, dynamic management will be implemented to reach a higher degree of customer-orientation and satisfaction.

There is still much work about this approach. Presently we have implemented a prototype system about model verification (MV), and automatic composition (AC) will be our future work. It is also important to make the formal description conveniently or automatically, which may finally determine if the approach can be widely applied. The possible way is to implement some automatic transformation from some existing or maturing standard description languages such as OWL/RDF.

REFERENCES