ENTERPRISE SYSTEM DEVELOPMENT WITH INVARIANT PRESERVING

A Mathematical Approach by the Homotopy Lifting and Extension Properties

Kenji Ohmori
Faculty of Computer and Information Sciences, Hosei University, Koganei-shi,Tokyo184-8584, Japan

Tosiyasu L. Kunii
Morpho, Inc. The University of Tokyo, Entrepreneur Plaza 5F, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

Keywords: Homotopy, HLP, HEP, π-calculus, Abstraction hierarchy, Invariant preserving.

Abstract: In this paper, a theoretical method for developing enterprise systems represented by the π-calculus is introduced. The method is based on the modern mathematics of homotopy theory. The homotopy lifting and extension properties are applied to developing systems in bottom-up and top-down ways with the incrementally modular abstraction hierarchy, where system development is carried out by climbing down abstraction hierarchy with adding invariants linearly. It leads to avoid combinatorial explosions causing an enormous waste of time and cost on testing. The system requirements and a state transition diagram drive the actions of an event by applying the HEP. Then, the state transition diagram and actions bring π-calculus processes by applying the HLP. These processes do not need testing because of invariant preserving.

1 INTRODUCTION

Why are tests necessary in the development process of enterprise systems? The Rational Unified Process (RUP) widely used as an iterative software development process framework includes tests as engineering disciplines. The designers preferably using the RUP believe that tests are inevitable, while knowing that much time and cost are spent on tests. When developing systems by the RUP, system requirements are modeled using the Unified Modeling Language (UML) consisting of several kinds of diagrams including class, use-case and activity diagrams. The individual-dependent and non-theoretical professional work from system requirements to UML diagrams leads to the necessity of tests. Activity diagrams and class diagrams are naturally different when designers are different. How do you verify theoretically that these different diagrams meet the system requirements? Without showing the correctness of modeling process, tests are required as the last tools for verification.

Invariants and invariant preserving are the most important concepts in science. By defining the most abstract invariants first and adding less abstract or more specific invariants step by step with preserving the previously added invariants, the correctness of modeling process is guaranteed. The incrementally modular abstraction hierarchy (IMAH) has been introduced in these papers(Kunii, 2005), (Kunii and Ohmori, 2006), (Ohmori and Kunii, 2006), (Ohmori and Kunii, 2007a), (Ohmori and Kunii, 2007b), (Ohmori and Kunii, 2008a), (Ohmori and Kunii, 2008b), where the IMAH has seven abstraction levels from the homotopy level to the set theoretical level, topological space level, adjunction space level, cellular space level, presentation level and view level. Invariants are added linearly while climbing down the abstraction hierarchy. The IMAH has been applied for architecture and modeling of cyberworlds.

On the way of climbing down the IMAH, methods of top-down, bottom-up or mixture of them are required for architecture and modeling of systems. The homotopy lifting property (HLP) and homotopy extension property (HEP) give theoretical backgrounds (Havey, 2005), (Sieradski, 1992), (Spanier, 1966) for a top-down and bottom-up method, respectively. Difficult problems can be solved by a divide and conquer

Ohmori K. and L. Kunii T. (2009).
ENTERPRISE SYSTEM DEVELOPMENT WITH INVARIANT PRESERVING - A Mathematical Approach by the Homotopy Lifting and Extension Properties.
In *Proceedings of the 11th International Conference on Enterprise Information Systems - Information Systems Analysis and Specification*, pages 116-123
DOI: 10.5220/0001981501160123

method in computer science, where a difficult problem is divided into two simple problems. The HLP consists of a total, base and lifting space, where the total space is complicated and is obtained in a bottom-up way from two simple spaces of the base and lifting space. As the HEP is dual to the HLP, the HEP gives a top-down way.

In this paper, the energy purchase problem is designed from its specification to a model represented by the π-calculus (Milner, 1999), (Sangiorgi and Walker, 1999), (Hennessy, 2001) using the HLP and HEP while climbing down the IMAH from the abstract level of the system requirements to the specific level of a π-calculus model.@It is shown that the most important concepts in this paper, that is, invariants and invariant preserving are formally and theoretically described by modeling the energy purchase problem (Havey, 2005) so that enterprise system development is changed from a hand-made professional job with testing to a theoretical and intelligent work without testing.

2 MATHEMATICAL BACKGROUNDS FOR DEVELOPING ENTERPRISE SYSTEMS

2.1 The Incrementally Modular Abstraction Hierarchy

The IMAH has the following levels starting from the most abstract to the most specific: the homotopy level; the set theoretical level; the topological space level; the adjunction space level; the cellular space level; the presentation level and the view level. On the homotopy level, the most abstract invariants with homotopy equivalence are defined using the HLP or HEP. On the set theoretical level, sets of spaces, which are mostly discrete, are defined with logical operations. On the topological space level, important invariants with isomorphism is introduced. On the adjunction level, invariants presenting the properties of dynamic changes, which are very important characteristics for organizing the structure of information, are defined using adjunction mapping. On the cellular space level, abstract physical structures such as frameworks of state transition diagrams are represented. On the presentation level, which is the starting point in the traditional architecture and modeling, designing entities such as UML diagrams and concrete state transition diagrams are defined. On the view level, program codes including π-calculus processes are obtained.

2.2 The Homotopy Lifting Property and Homotopy Extension Property

The mathematical backgrounds for the HLP and HEP are summarized as follows.

Def. 1: Continuous maps p, q are homotopic if there exists a continuous map $H : X \times I \to Y$ such that $H(x,0) = p(x)$ and $H(x,1) = q(x)$, where I is the unit interval $[0,1]$. H is called homotopy of p and q, denoted by $p \simeq q$.

Def. 2: A continuous map $\lambda : I \to X$ yields a path. $\lambda(0) = x$ and $\lambda(1) = y$ are called the initial and terminal points. The path is denoted by $w = (W,\lambda)$ where $W = \lambda(I)$.

Def. 3: A fiber bundle is a quadruple $\xi = (E,B,F,p)$ consisting of a total space E, a base space B, a fiber F, and a bundle projection that is a continuous surjection called $F-$bundle $p : E \to B$ such that there exists an open covering $\mathcal{U} = \{U\}$ of B and, for each $U \in \mathcal{U}$, a homeomorphism called a coordinate chart $\varphi_U : U \times F \to p^{-1}(U)$ exists such that the composite $U \times F \to p^{-1}(U) \to U$ is the projection to the first factor U. Thus the bundle projection $p : E \to B$ and the projection $p_B : B \times F \to B$ are locally equivalent. The fiber over $b \in B$ is defined to be equal to $p^{-1}(b)$, and we note that F is homeomorphic to $p^{-1}(b)$ for every $b \in B$, namely $\forall b \in B, F \cong p^{-1}(b)$.

Def. 4: Given any commutative diagram of continuous maps as shown in Fig. 1, the map $p : E \to B$ has the homotopy lifting property if there is a continuous map $\hat{H} : Y \times I \to E$ such that $\hat{H} \times i_0 = h$ and $p \circ \hat{H} = H$. The homotopy $\hat{H}$ thus lifts H through p and extends h over i_0 where $i_0(a) = (a,0)$.

Def. 5: A fibration is a continuous map $p : E \to B$ that has the homotopy lifting property.The homotopy extension property is dual to the homotopy lifting property. The homotopy extension property is defined as follows.

Def. 6: Given any commutative diagram of continuous maps as shown in Fig. 2, there is a continuous map $\hat{K} : X \to Y^I$ such that $p_0 \times \hat{K} = k$ and $\hat{K} \times i = K$. The homotopy $\hat{K}$ thus extends K over i and lifts k through p_0 where $p_0(\lambda) = \lambda(0)$.

Def. 7: An inclusion of a closed subspace $i : A \hookrightarrow X$ is a cofibration if i has the homotopy extension property. Y^I is the path space on Y. The path space is defined as follows.

Def. 8: The path space on X, denoted X^I, is the space $\{\lambda : I \rightarrow X | continuous\}$ endowed with the compact-open topology.

Def. 9: Let us start with a topological space X and attach another topological space Y to it. Then, $Y_f = Y \sqcup_f X = Y \sqcup X/\sim$ is an adjunction space obtained by adjuncting Y to X by an adjunction map, f (or by identifying each point $y \in Y_0 \mid Y_0 \subset Y$ with its image $f(y) \in X$ by a continuous map f). $\sqcup$ denotes a disjoint union. The adjuncting map f is a continuous map such that $f : Y_0 \rightarrow X$, where $Y_0 \subset Y$. Thus, the adjuncting space $Y_f = Y \sqcup X/\sim$ is a case of quotient spaces
$Y \sqcup X/\sim = Y \sqcup_f X = Y \sqcup X/(x \sim f(y) \mid \forall y \in Y_0)$.

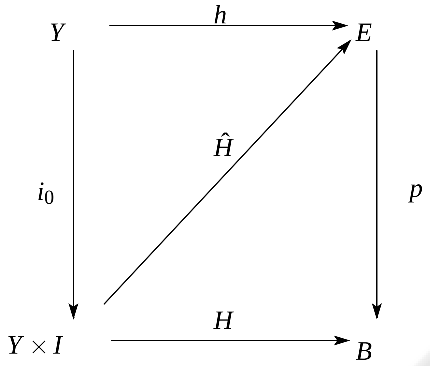


Figure 1: Homotopy Lifting Property.

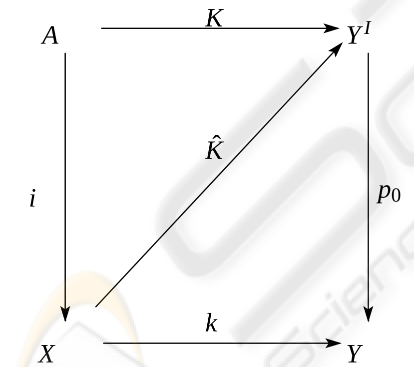


Figure 2: Homotopy Extension Property.

3 APPLYING THE HOMOTOPY THEORY TO AN ENERGY PURCHASE PROBLEM

3.1 An Energy Purchase Problem

The energy purchase@problem is shown in the book (Havey, 2005). The outline of this problem is as follows. A customer is registerd to an energy supplier. The customer ordinarily buys energy from the energy supplier. Because of deregulation, the customer is allowed to buy energy from a retailer. As there are several retailers, the custome is also allowed to switch retailers. When the customer wants to switch retailers, he has to notify the energy supplier of a retailer switch and the energy supplier reports this switch to the retailers. The customer also switches back to the energy supplier

3.2 The π-calculus Process

The π-calculus is a parallel processing model where communication links are dynamically changed. The action prefixes π are a generalization of actions. An action prefix expresses either sending or receiving a message or making a silent transition. The π-prefixes are defined as follows.

$$\begin{aligned} \pi \; &::= \; x(y) \; \textit{receive y along x} \\ &::= \; \overline{x}(y) \; \textit{send y along x} \\ &::= \; \tau \; \textit{unobservable action} \end{aligned}$$

The set P^π of π-calculus process expressions is defined as follows:

$$P ::= 0 \mid \sum_{\lambda \in \Lambda} \pi_\lambda . P_\lambda \mid P_1 | P_2 \mid new\ a\ P \mid !P$$

0 is an inaction process that can do nothing. The processes $\sum_{\lambda \in \Lambda} \pi_\lambda . P_\lambda$ are called sums. Each item is a process and only one item is executed. Λ is any finite indexing set. In a sum $\sum_{\lambda \in \Lambda} \pi_\lambda . P_\lambda$, it is said that P_λ is guarded by π_λ since the action by π_λ has to proceed before P_λ becomes active. $P_1 | P_2$ can proceed independently and interact by shared names. *new a P* restricts the scope of the name *a* to *P*. $!P$ repeats P infinitely.

Two examples are given for explaining the π-calculus. The first example $\overline{x}(y).P_1 | x(z).P_2$ shows how a message is sent via a communication link. $x(y)$ of an action prefix π can receive any name and y is bounded to the name or substituted by the received name. $\overline{x}(y).P_1 | x(z).P_2$ executes two processes. $\overline{x}(y).P_1$, where x is used as a communication link, sends y via x. $x(z).P_2$ receives y via x where z is substituted by y.

The second example $\overline{x}(y).P_1 | x(z).\overline{z}(u).P_2 | y(w).P_3$ shows how a communication link is dynamically changed. The first process notifies the second process of a communication link name, which will be used by the second process for communicating with the third process. That is, $\overline{x}(y).P_1$ and $x(z).\overline{z}(u).P_2$ changes a message like the above explanation. $x(z).\overline{z}(u).P_2$ receives y via x, it sends u via y since z is bounded to y. Finally, $y(w).P_3$ receives u via y.

3.3 A Hep Application

The energy purchasing problem is solved by the following steps.

1) A state transition diagram is obtained by climbing down from the homotopy level to the presentation level.

2) Using the HEP, the actions for an event causing a situation change are obtained from the system requirements and the state transition diagram. The actions are also generated by climbing down the abstraction hierarchy.

3) Using the HLP, the processes for the energy purchase problem are obtained.

At first, the system requirements are formed as a topological space. Assuming that there are one customer, one energy supplier and two suppliers without losing generality, the system requirements consist of the following elements.

x_1: The customer ordinarily buys energy from the energy supplier.

x_2: The customer can buy energy from a retailer.

x_3: When the customer changes his purchasing place from the supplier to a retailer, he has to notify the supplier of his enrollment.

x_4: When the customer changes his purchasing place from a retailer to another retailer, he has to notify the supplier of his switch.

x_5: When a customer changes his purchasing place from a retailer to the supplier, he has to notify the supplier of his drop.

The set of these elements forms a topological space by introducing discrete topology.

A state transition diagram is constructed by extracting states and transitions from the system requirements. This process is achieved by providing two abstract states, e_1^0 for purchase on standard supply and e_2^0 for purchase by taking advantage of deregulation and three state transitions, e_1^1 from e_1^0 to e_2^0 caused by an enrollment, e_2^1 from e_2^0 to e_1^0 by a drop and e_3^1 from e_2^1 to e_1^0 by a switch. As states and transitions are ordinary represented as vertexes and edges, e_1^0 and e_2^0 are defined as 0-dimensional spaces and e_1^1, e_1^2 and e_2^3 as 1-dimensional spaces. The set of $\{e_1^0, e_2^0, e_1^1, e_2^1, e_3^1\}$ is formed as a topological space by introducing discrete topology. States and transitions are connected to form a state transition diagram by applying adjuncting maps. For this problem, both boundaries of ∂e_1^1 are attached to e_1^0 and e_2^0. In the same way, the boundaries of other state transitions are attached to the corresponding states.

To form the state transition diagram as a CW-complex, two 2-dimentional spaces: e_1^2 and e_2^2, are provided and these boundaries are attached to the corresponding edges. The CW complex is further transformed to a more specific state transition diagram on the presentation level as shown in Fig. 3. The state transition diagram is a subset of the system requirements where $e_1^0, e_2^0, e_1^1, e_2^1$ and e_3^1 correspond to x_1, x_2, x_3, x_5 and x_4, respectively.

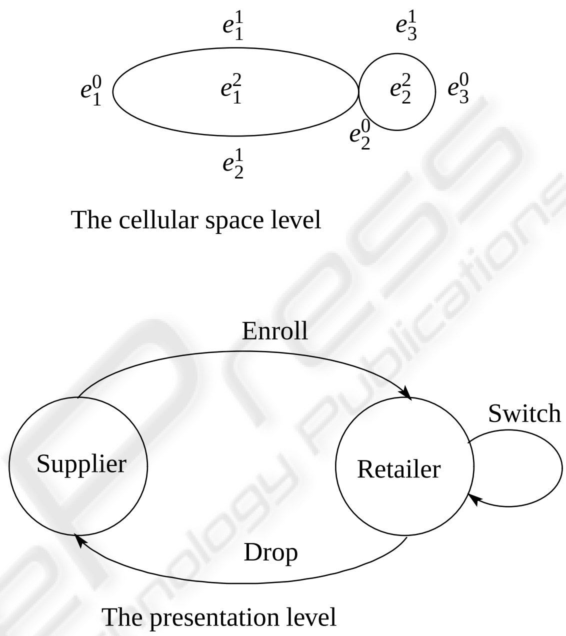


Figure 3: State transition diagram.

The energy purchasing problem has three situations: Z_1 for purchasing from the energy supplier, Z_2 for purchasing from retailer A and Z_3 for purchasing from retailer B. Sometimes, a situation is called a use-case. As the customer is allowed to change a purchasing place, the sequence of situations generated by changing purchase places is represented in the form of $(Z_1(Z_2|Z_3)^*)^*$. One example sequence is shown from Fig. 4 to Fig. 7. The change of a purchasing place is caused by events such as an enrollment, switch and drop. Each of them has a series of actions.

Now, let's consider obtaining actions in an event using the HEP. Later, the actions are transformed into π-processes. In the HEP, X is the system requirements and A is the state transition diagram. Y is a series of actions for an event. There are three events: enrollment E_1, switch E_2 and drop E_3. E_1, E_2 and E_3 cause transitions of $Z_1 \to Z_2|Z_1 \to Z_3$, $Z_2 \to Z_3|Z_3 \to Z_2$ and $Z_2 \to Z_1|Z_3 \to Z_1$, respectively. Each event has a series of actions such that $Y_1 = Y \times E_1$ has {1) *Mike* notifies *Sup* of the purchase enrollment to *RetA*, 2) *Sup* receives from *Mike* the purchase enrollment to *RetA*, 3) *Sup* reports to *RetA* the purchase enrollment of *Mike*, 4) *RetA* receives from *Sup* the pur-

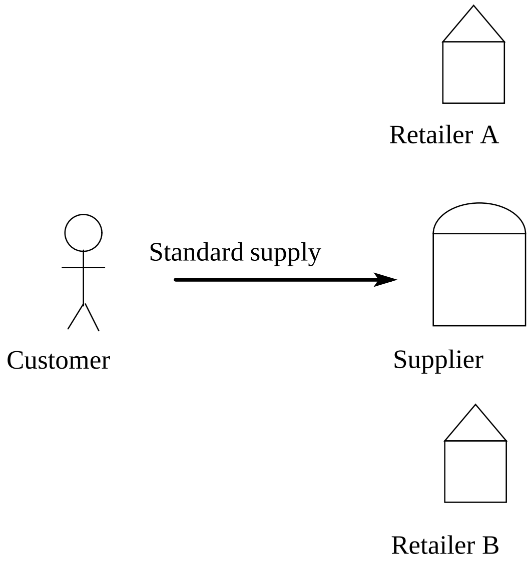


Figure 4: Customer is not enrolled with a retailer, on standard supply.

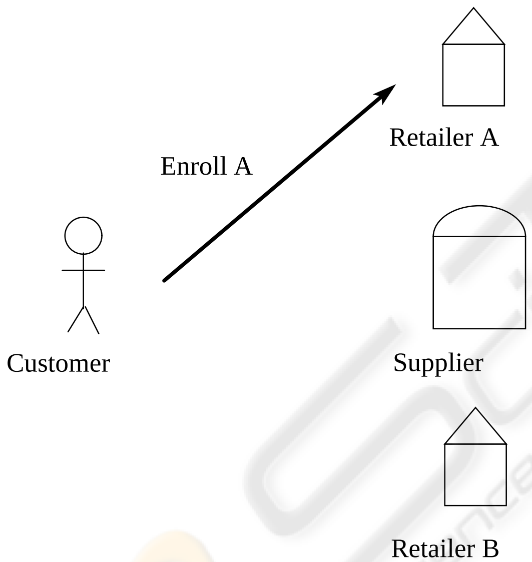


Figure 5: Customer enrolls with retailer A.

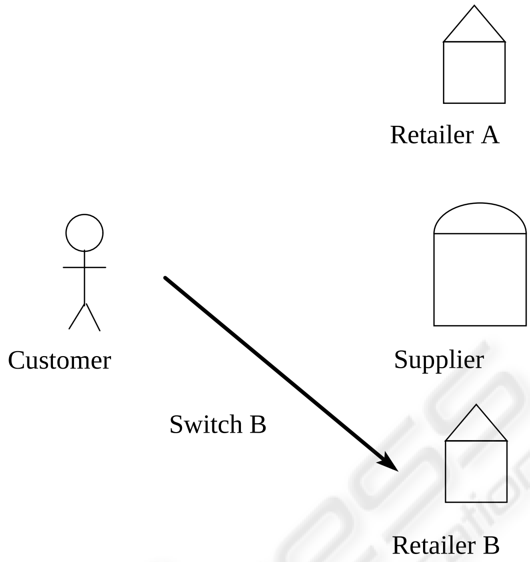


Figure 6: Customer switches to retailer B.

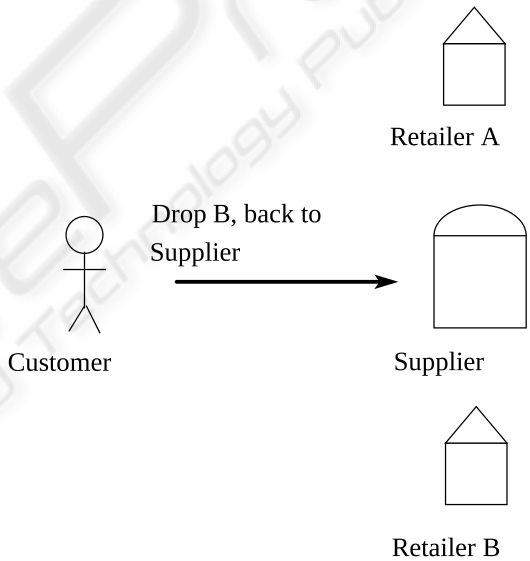


Figure 7: Customer drops retailer, back to standard supply.

chase enrollment of *Mike*} where the customer, the energy supplier and two retailers are represented by *Mike*, *Sup*, *RetA* and *RetB*. Actions are obtained from the system specification. In the above case, x_3 defines its actions.

For each event, the following actions are obtained.

1. Event: Y_1 (Purchase change from the energy supplier to a retailer)
   1) Sub event: Y_{11} (Change from *Sup* to *RetA*)
   y_{11}^1: *Mike* notifies *Sup* of a purchase enrollment to *RetA*.
   y_{11}^2: *Sup* receives from *Mike* a purchase enrollment to *RetA*.
   y_{11}^3: *Sup* reports to *RetA* a purchase enrollment of *Mike*.
   y_{11}^4: *RetA* receives from *Sup* a purchase enrollment of *Mike*.
   2) Sub event: Y_{12} (Change from *Sup* to *RetB*)
   y_{12}^1: *Mike* notifies *Sup* of a purchase enrollment to *RetB*.
   y_{12}^2: *Sup* receives from *Mike* a purchase enrollment to *RetB*.
   y_{12}^3: *Sup* reports to *RetB* a purchase enrollment of *Mike*.
   y_{12}^4: *RetB* receives from *Sup* a purchase enrollment of *Mike*.

2. Event: Y_2 (Purchase change among retailers)
   1) Sub event: Y_{21} (Change from *RetA* to *RetB*)
   y_{21}^1: *Mike* notifies *Sup* of a purchase switch from *RetA* to *RetB*.
   y_{21}^2: *Sup* receives from *Mike* a purchase switch from *RetA* to *RetB*.
   y_{21}^3: *Sup* reports to *RetA* a purchase drop of *Mike*.
   y_{21}^4: *Sup* reports to *RetB* a purchase enrollment of *Mike*.
   y_{21}^5: *RetA* receives from *Sup* a purchase drop of *Mike*.
   y_{21}^6: *RetB* receives from *Sup* a purchase enrollment of *Mike*.
   2) Sub event: Y_{22} (Change from *RetB* to *RetA*)
   y_{22}^1: *Mike* notifies *Sup* of a purchase switch from *RetB* to *RetA*.
   y_{22}^2: *Sup* receives from *Mike* a purchase switch from *RetB* to *RetA*.
   y_{22}^3: *Sup* reports to *RetB* a purchase drop of *Mike*.
   y_{22}^4: *Sup* reports to *RetA* a purchase enrollment of *Mike*.
   y_{22}^5: *RetB* receives from *Sup* a purchase drop of *Mike*.
   y_{22}^6: *RetA* receives from *Sup* a purchase enrollment of *Mike*.

3. Event: Y_3 (Purchase change from the retailer to the energy supplier)
   1) Sub event: Y_{31} (Change from *RetA* to *Sup*)
   y_{31}^1: *Mike* notifies *Sup* of a purchase drop out of *RetA*.
   y_{31}^2: *Sup* receives from *Mike* a purchase drop out of *RetA*.
   y_{31}^3: *Sup* reports to *RetA* a purchase drop of *Mike*.
   y_{31}^4: *RetA* receives from *Sup* a purchase drop of *Mike*.
   2) Sub event: Y_{32} (Change from *RetB* to *Sup*)
   y_{32}^1: *Mike* notifies *Sup* of a purchase drop out of *RetB*.
   y_{32}^2: *Sup* receives from *Mike* a purchase drop out of *RetB*.
   y_{32}^3: *Sup* reports to*RetB* a purchase drop of *Mike*.
   y_{32}^4: *RetB* receives from *Sup* a purchase drop of *Mike*.

Let's consider constructing Y^I. The interval I is represented by $(E_1E_2^*E_3)^*$. Y^I is a set of paths, each of which is the sequence of the i^{th} action along I such that

$$\begin{aligned} Y^I \;=\{ \;& ((y_{11}^1|y_{12}^1)(y_{21}^1|y_{22}^1)^*(y_{31}^1|y_{32}^1))^*, \\ & ((y_{11}^2|y_{12}^2)(y_{21}^2|y_{22}^2)^*((y_{31}^2|y_{32}^2))^*, \ldots\}. \end{aligned}$$

However, a path yielded by Y^I is not allowed to include $y_{1m}^k y_{2n}^k$and $y_{2m}^k y_{2n}^k$ where $m \neq n$, and $y_{1m}^k y_{3n}^k$and $y_{2m}^k y_{3n}^k$ where $m = n$ as it part.

It is not hard to show that A is included in X, K and $\hat{K}$ are homotopies and $\hat{K}$ extends K over i.

3.4 A HLP Application

Now, consider obtaining π-calculus processes of the energy purchase problem using the HLP where B and $Y \times I$ are described as a state transition diagram and events, and π-calculus processes are accommodated in E. E consists of layers for agents in the energy purchasing problem, where the customer, supplier and customers are agents. When solving this problem, the agent for the customer is categorized into two types: the customer purchasing on standard supply and the customer purchasing from a retailer. The obtained processes are shown from Fig. 8 to Fig. 11, where there are four processes: P_1 for the supplier, P_2 for retailers, P_3 for the customer on standard supply and P_4 for the customer purchasing from retailer. Each process is represented in the same CW-complex on the cellular space level as the state transition diagram.

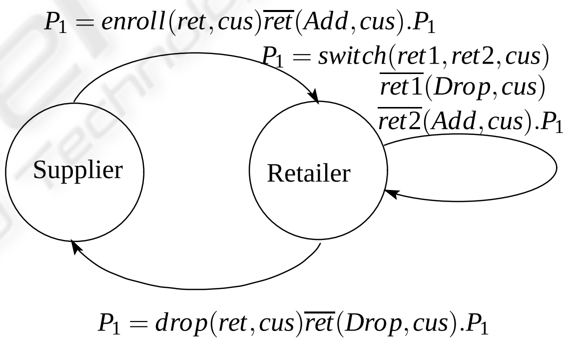


Figure 8: Process for the supplier.

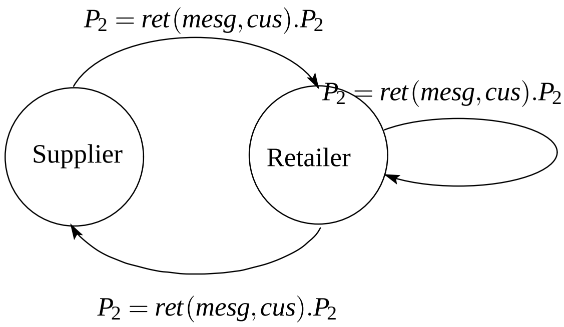


Figure 9: Process for the retailers.

$$P_3 = \overline{enroll}(RetA, Mike).P_4(RetA) + \overline{ebroll}(RetB, Mike).P_4(RetB)$$

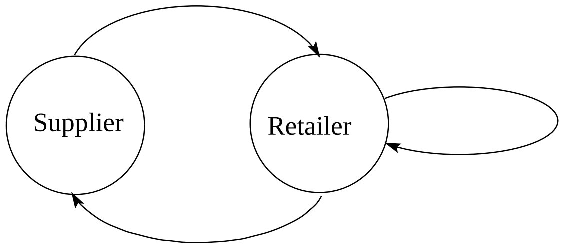
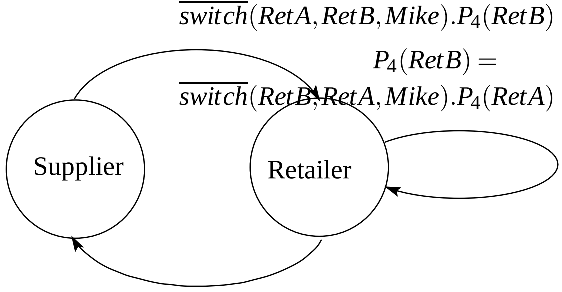


Figure 10: Process for the customer on standard supply.

$$P_4(RetA) = \overline{switch}(RetA, RetB, Mike).P_4(RetB)$$

$$P_4(RetB) = \overline{switch}(RetB, RetA, Mike).P_4(RetA)$$



$$P_4(RetA) = \overline{drop}(RetA, Mike).P_3$$

$$P_4(RetB) = \overline{drop}(RetB, Mike).P_3$$

Figure 11: Process for the customer purchasing from a retailer.

However, these are different on the presentation level. Arcs of the presentation level are obtained from the events as follows. Firstly, Y_1 is mapped into these state diagrams. As Y_1 has two subspaces Y_{11} and Y_{12}, both of them are mapped step by step in the following way. y_{11}^1 that is an element of Y_{11} is mapped as a process $\overline{enroll}(RetA, Mike)$. $P4(RetA)$ for the customer purchasing on standard supply. y_{11}^2 and y_{11}^3 are mapped as a process $enroll(ret, cus).\overline{ret}(Add, cus)$. $P1$ for the supplier. y_{11}^4 is mapped as a process $ret(message, cus)$. $P2(ret)$ for the retailer. Other mappings are carried out in the same way for all events. It is not also hard to prove that H and $\hat{H}$ are homotopies and $\hat{H}$ lifts H through p.

4 REFINEMENT STEPS

A state diagram is a connected CW-complex that is homotopy equivalent to a point. If a state transition diagram is reduced to a point, then the processes distributed on arcs are assembled into a single expression. The processes for the supplier, retailers, customer on standard supply and customer purchasing from a retailer are obtained by the following expressions.

$$\begin{aligned}
P_1 &= enroll(ret, cus)\overline{ret}(Add, cus) \\
&\quad + switch(ret1, ret2, cus)) \\
&\quad \overline{ret1}(Drop, cus)\overline{ret2}(Add, cus) \\
&\quad + drop(ret, cus)\overline{ret}(Drop, cus)).P_1. \\
P_2 &= ret(mesg, cus).P_2. \\
P_3 &= \sum_{ret \in \{RetA, RetB\}} \overline{enroll}(ret, Mike).P_4(ret). \\
P_4(ret1) &= \sum_{ret2 \in \{RetA, RetB\} \cap (ret1 \neq ret2)} \\
&\quad \overline{switch}(ret1, ret2, Mike).P_4(ret2) \\
&\quad + \overline{drop}(ret1, Mike).P_3
\end{aligned}$$

where $ret1 \in \{RetA, RetB\}$.
The generalized expression is also obtained using the HLP. When the problem is extended to have several customers and retailers, the above processes are generalized from the HLP as follows, where *Cus* and *Ret* are the sets of the customers and retailers and *cus* and *ret*1 are elements of *Cus* and *Ret*, respectively.

$$\begin{aligned}
P_3(cus) &= \sum_{ret \in Ret} \overline{enroll}(ret, cus).P_4(cus, ret). \\
P_4(cus, ret1) &= \sum_{ret2 \in Ret \cap (ret1 \neq ret2)} \\
&\quad \overline{switch}(ret1, ret2, cus).P_4(cus, ret2) \\
&\quad + \overline{drop}(ret1, cus).P_3(cus).
\end{aligned}$$

5 CONCLUSIONS

A new development method using homotopy theory has been introduced. The HEP and HLP are the most abstract and powerful tools for categorizing mathematical objects. Homotopy equivalence are maintained within the HEP and HLP. In this paper, the HEP and HLP have been introduced for establishing fundamentals in software science. Architecture, modeling and designing are performed by climbing down the abstraction hierarchy adding invariants incrementally using the HEP and HLP. The IMAH has 7 levels from the homotopy space level to the set theoretical level, the topological space level, the adjunction level, the cellular space level, presentation level and the view level. The system requirements for the energy purchase problem on the set theoretical level is transformed to the π-calculus processes on the view level using the HEP and HLP. While climbing down the abstraction hierarchy, invariants are incrementally added from abstract to specific ones while keeping homotopy equivalence on all levels and topological equivalence on less abstract levels than topological space level. The invariant preserving by homotopy equivalence and topological equivalence makes test unnecessary and enables to avoid combinatorial explosions that the conventional method is facing now.

The HEP and HLP have been introduced for a top-down and bottom-up design, respectively. This is a new application field of the HEP and HLP. When solving the energy purchase problem, the state transition diagram and events, which are specific modeling and designing entities from the system requirements on the set theoretical level, have been obtained using the HEP in a top-down way, where the system requirements, state transition diagram and events are homotopy equivalent. Then, the π-calculus processes on the view level have been obtained from the state transition diagram and events using the HLP in a bottom-up way. The introduced method gives a mathematical foundation for a top-down and bottom-up design, which are carried out by individual know-how in the conventional system. The top-down and bottom-up design also keeps invariants with making tests necessary.

The development steps introduced here start from the system requirements, move to obtain the state transition diagram and actions for events and end with the π-calculus processes. These operations are automatic. By developing a development framework along these steps, the development of enterprise systems will be automated from the system requirements to its BPEL codes.

This paper has constructed a bridge between computer science and the modern mathematics of homotopy theory. The introduced method is general and applicable to any fields of computer science including computer graphics, computer architecture, network architecture and cyberworlds. In this paper, the π-calculus is used to express processes for enterprise systems. It is easily applicable to communicating sequential processes, which is another process algebra and powerful tools in the field of embedded systems.

REFERENCES

Havey, M. (2005). *Essential Business Process Modeling*. O'Reilly Media, Inc, Cambridge.

Hennessy, M. (2001). *A Distributed Pi-Calculus*. Cambridge University Press, Cambridge.

Kunii, T. L. (2005). Cyberworlds -theory, design and potetial-. *The Transactions of The Institute of Electronics, Information and Communication Engineers*, E88-D(5):790–800.

Kunii, T. L. and Ohmori, K. (2006). Cyberworlds: Architecture and modeling by an incrementally modular abstraction hierarchy. *The Visual Computer*, 22(12):949–964.

Milner, R. (1999). *Communicating And Mobile Systems: Pi-Calculus*. Cambridge University Press, Cambridge.

Ohmori, K. and Kunii, T. L. (2006). An incrementally modular abstraction hierarchy for linear software development methodology. *Int. Conf. on Cyberworlds 2006*, pages 216–223.

Ohmori, K. and Kunii, T. L. (2007a). Development of an accounting system. *ICEIS2007*, pages 437–444.

Ohmori, K. and Kunii, T. L. (2007b). The mathematical structure of cyberworlds. *Int. Conf. on Cyberworlds 2007*, pages 100–107.

Ohmori, K. and Kunii, T. L. (2008a). Mathematical modeling of ubiquitous systems. *Int. Conf. on Cyberworlds 2008*, pages 69–74.

Ohmori, K. and Kunii, T. L. (2008b). A pi-calculus modeling method for cyberworlds systems using the duality between a fibration and a cofibration. *Int. Conf. on Cyberworlds 2008*, pages 363–370.

Sangiorgi, D. and Walker, D. (1999). *The Pi-Calculus: A Theory of Mobile Processes*. Cambridge University Press, Cambridge.

Sieradski, A. J. (1992). *An introduction to topology and homotopy*. PWS-Kent Publishing Company, Boston.

Spanier, E. H. (1966). *Algebraic topology*. Springer-Verlag, New York.