Chapter 30

MAPPING "ENTERPRISE BUSINESS ARCHITECTURE" TO "INFORMATION SYSTEMS FRAMEWORK"

Atsushi Yamaguchi, Motoyuki Suzuki, Masanori Kataoka
Hitachi INS Software

Abstract: Globalization accelerates worldwide Business Modularization, and Business Modularization cannot be achieved without the help of Internet. In this paper, we will clarify the role of Internet in Business Modularization, and will discuss the systematic methodology to leverage Enterprise Business Architecture, which is placed at the origin of Information Systems Development Lifecycle.

Key words: Enterprise Business Architecture, Business Modularization, Information Systems Framework

1. INTRODUCTION

1.1 Business Modularization

Fast-paced globalization of business has forced enterprises to categorize their business elements, the process which is referred to as "Business Modularization". Enterprises concentrate their efforts on their core business and sell their core products and services worldwide. On the contrary as for the non-core business, enterprises procure them from or outsource them to their international partners.

Examples of Business Modules are shown in Table 1.

Table 1. Examples of Business Module

Module	Example
People	Human Organization, Service Package
Things	Products, Parts
Money	Financial Package
Knowledge/Information	Knowledgeware, Information Contents

1.2 Outline of Methodology

In this paper, we will place "Enterprise Business Architecture" at the origin of Information Systems Development Lifecycle, and will discuss the systematic techniques to leverage this architecture in system design phase.

Figure 1 illustrates the outline of the methodology.

There are two achievements that are Enterprise Business Architecture in business analysis phase and Information Systems Framework in system design phase. Our challenge is to bridge the gap between Enterprise Business Architecture and Information Systems Framework. Information Systems Framework is build by leveraging Enterprise Business Architecture.

The outline of methodology to build Enterprise Business Architecture, the outline of methodology is:

(1) To clarify the Competitive Strategy.

(2) To make Business Model Transformation with the above strategy.

The result of transformation is important for Information System Framework.

The outline of methodology to build Information Systems Framework, the outline of methodology is:

(1) To build IT system with PIM (Platform Independent Model)

(2) To transform PIM to PSM (Platform Specific Model)

2. ENTERPRISE BUSINESS ARCHITECTURE

Enterprise Business Architecture consists of Business Competitive Strategy and Business Systems Architecture. In addition, Business Systems Architecture consists of Business Module and Inter-Module Structural Operator.

Business Module is the basic element of business such as "People", "Things", "Money" and "Knowledge/Information". Every Business System contains Business Modules. Well-balanced structure of "People", "Things" and "Money" is important, and the structure should be well armed with

"Knowledge/Information". Relationship between Business System and Business Modules is shown in *Figure 2.*

Business Module is referred to as *Module* in this paper.

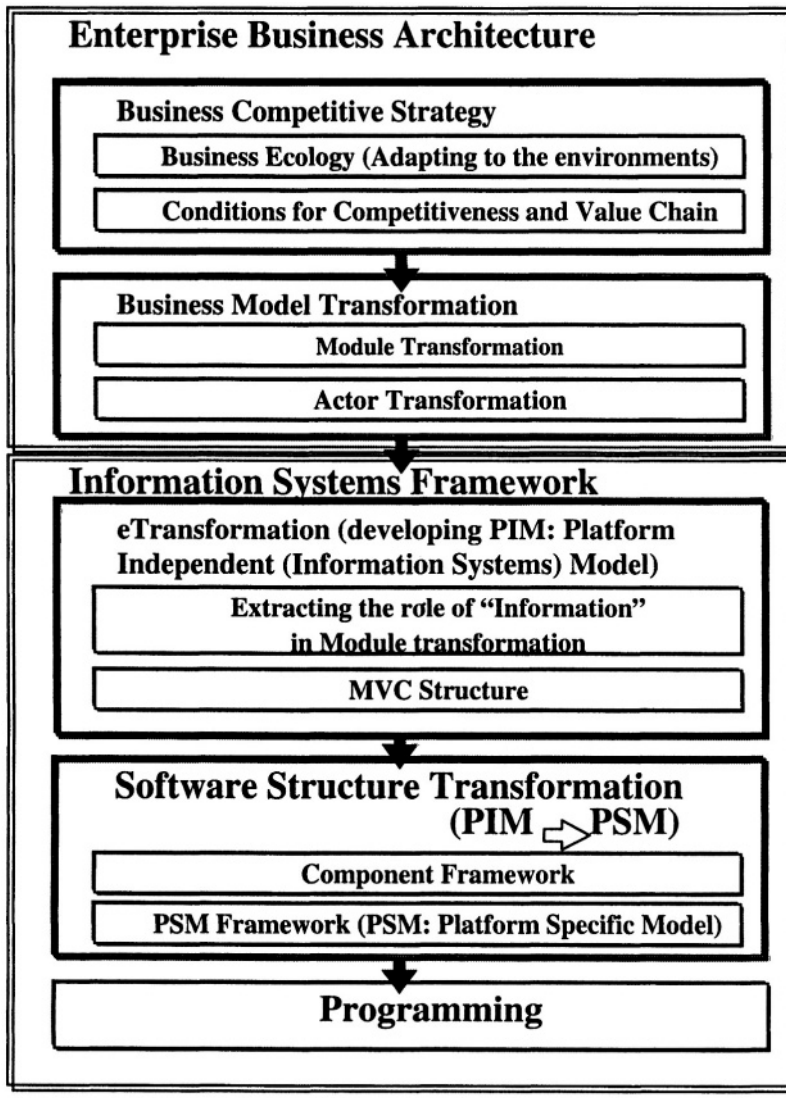


Figure 1. Methodology

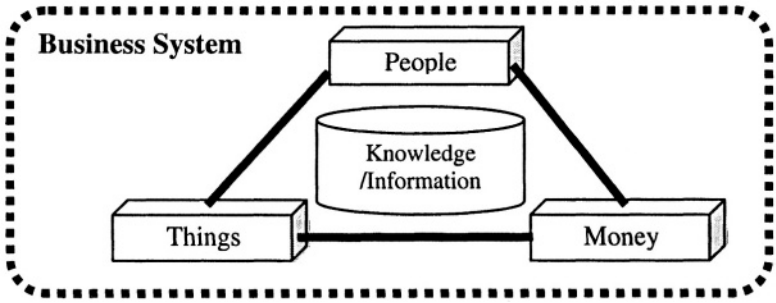


Figure 2. Relationship between Business System and Business Modules

2.1 Business Competitive Strategy

2.1.1 Business Ecology

Business Ecology is strategy for an enterprise to become competitive and to adapt itself to the environment, which is based on the idea that enterprise as a living being survives with suppliers, customers/consumers, and social environments. Factors that influence the enterprises are diversity, interaction, reciprocal action and natural selection. Globalization of recent years is strengthening the effect of these Influential factors. Environmental elements are Social Environment, Customer/Consumer Environment and Value Chain with the Suppliers as shown in Table 2.

Table 2. Environmental Elements

Environmental Elements	Example
Social Environment	National Organization, Politics, Geographical Feature, Economy, Culture, Labor
Customer/Consumer Environment	Value for the Customer/Consumer, Differentiation of Product Quality, Information "Contents" added to the Products
Value Chain with the Suppliers	Cost of Materials/Parts/Products, Quality/Function of Products, Selection/Fostering of Suppliers

2.1.2 Conditions for Competitiveness

The following are the 4 conditions to become competitive:

(1) Zoning, Maintaining territory

(2) Adaptation and Evolution in the Environment
(3) Flexibility of the organization to be adaptive
(4) Survival of the fittest, Natural selection

Parameters to become competitive are Resource Capacity, Scale, Internal Control, Scarcity, Variety of supply source, Market Structure and Transition as shown in Table 3.

Table 3. Parameters to become competitive

Parameters for competitiveness	Example
Resource Capacity	People, Things, Money, Knowledge/Information
Scale	Expansion, Reduction
Internal Control	Concentration, Distribution
Scarcity	Place, Position, Resource, Knowledge
Variety of supply source	Vendors
Market Structure	Symbiosis, Competition
Transition	Transition to a new domain

The following 3 sequences of procedures are important to map out the strategy to achieve competitiveness:

(A) To clarify external factors.
(B) To change internal factors depending on external factors.
(C) Design/Modify the Business Systems Architecture in order to change internal factors.

Examples of external factors are product quality, product price, amount of supply, scarcity, brand, CRM(Customer Relationship Management), competitors and market domain. Examples of internal factors are people (especially leaders), production facilities, production engineering, development (innovation) capacity, SCM(Supply Chain Management) and partners. Examples of module operations in the Business Systems Architecture to change internal factors are Module Transformation and Actor Transformation.

2.2 Business Systems Architecture

2.2.1 Characteristics of a module

Business Systems Architecture consists of *Module* and Inter-Module Structural Operator. It can be said that *Module* is the characterized box as shown in *Figure 3*.

External specification of *Module* consists of knowledge that can be expressed explicitly by language or sentence, in other words, a "protocol".

Meanwhile Internal structure of *Module* consists of subjective or physically acquired knowledge such as thoughts/views that are sometimes hard to express by language or sentence, in other words, "process".

It is important to keep the external specification of a *Module* stable and to keep the internal structure of a *Module* flexible to adapt to and survive in free competition environment.

Thus, advantages of Modularization are as follows:

(1) Modules are adaptive to rapid environmental change.
(2) Modules are divide jobs into independent parallel jobs.
(3) Only strong *Modules* with good internal structure survive the free competition between modules.

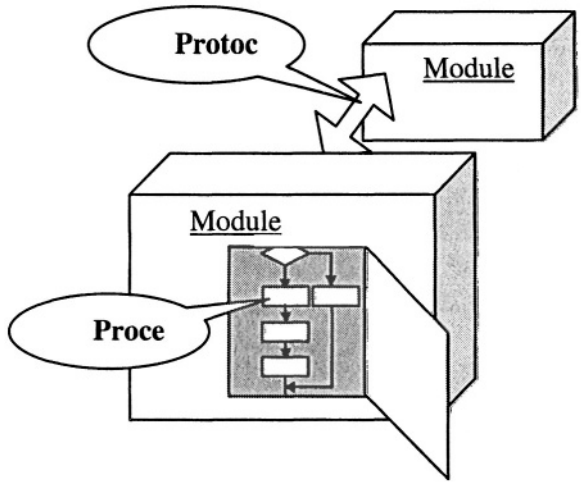


Figure 3. Characteristics of a Module

2.2.2 Inter-Module Structural Operator

Inter-Module Structural Operator illustrates how the connections between the modules are established. The formats of Inter-Module Structural Operators are Set, Hierarchy, Chain, Hub and Web. The Inter-Module Structural Operators are explained in Table 4.

While Web is important to energize the relationship among *Modules,* Hub is important to coordinate *Modules.* Many links are established between variety of enterprises and people to form a Web, and a Hub works as the central place to coordinate *Modules.* In the real world, we usually utilize both Hub and Web.

2.2.3 Module metrics

Module metrics consists of Module Strength, Module Coupling, Module Granularity and Degree of Encapsulation [2]. Module metrics are important to evaluate successfulness of Modularization.

Module Strength is a metric to evaluate module stability and reliability. When a module is evaluated in the perspective of module strength, it can be said that a module with high Functional Strength has high module strength and is highly independent, and a module with high Set Strength has low module strength and is highly dependent. Informational Strength positions itself in between Set Strength and Functional Strength, and it indicates that modules are related to some shared information. Module Strength is expected to be strong.

Module Coupling is a metric to evaluate dependency between *Modules*.

Table 6 illustrates Module Coupling metrics. As to Module Coupling, weaker coupling means self-organized and greater independence. Control coupling with complete dependence has no independence. *Modules* should have high independence and weak coupling.

Module Granularity is a metric to evaluate module size or module complexity. In case management resource is people, Module Granularity indicates the size of an organization unit or an enterprise, e.g. 10 people, 100 people, 1000 people and 10000 people. In case of things, Module Granularity is size or complexity of parts or products, e.g. m, km, g. In case of money, Module Granularities are amount of money for investment, development, etc, e.g. $1, $100 and $10,000. It is not necessarily appropriate to suggest that large *Module* or small *Module* is better.

Degree of Encapsulation is a metric to evaluate the degree of module encapsulation or capacity to provide good external interface. The measures are as follows:

(A) Single function/Multi Function

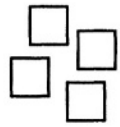
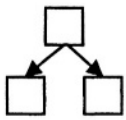
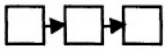
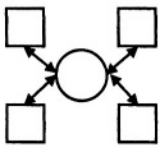
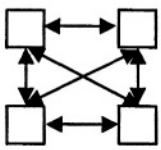
(B) With/Without memory

- With/without internal state
- With/without internal data

(C) Dependent/independent on platform

Higher degree of encapsulation is expected for *Modules* to be highly independent.

Table 4. Inter-Module Structural Operator

Operator	Figure	Description
Set		Set of Modules without Relation. e.g.: Parts stocked in warehouse.
Hierarchy		Hierarchy of Modules. e.g.: Hierarchical structure of physical parts.
Chain		Modules are chained to form a flow. e.g.: SCM (Supply Chain Management)
Hub		Modules are coordinated through a central place. The place could be physical place or abstract coordinator. cf. "Chain" could be treated as a kind of "Hub".
Web		Free relationship among modules. cf. "Web" is frequently used in open connection with extra-enterprise systems.

2.3 Business Transformation Operations

It is required for Enterprises to change their internal factor of Business Systems Architecture to adapt themselves to environmental (external factor) changes. Business model transformation is such an adaptation process that includes following module transformation operations.

(1) Augmenting new *Modules.*
(2) Excluding existing *Modules.*
(3) Dividing *Modules.*
(4) Integrating *Modules.*
(5) Substituting *Modules.* (Actor Transformation)
(6) Porting *Modules.* (to be common modules)
(7) Creating a new hierarchy by extracting common parts from multiple *Modules.*

Table 5 explains the Module Strength of each module.

Table 5. Module Strength Metrics

Strength		Weak ⟷ Strong		
Modules		Set Strength	Informational Strength	Functional Strength
People	Entity	Conglomerate	Grouped enterprises	Single function company (e.g. venture)
	Process	Set of variety of business processes	Information sharing among processes	Organization of professionals
Things	Entity	Set of independent elements	Set of related elements	Standard (Reusable) parts
	Process	Set of variety of processes	Concentrated management of process information	A single function process
Money	Entity	Set of variety of funds	Funds from market (e.g. stock funds)	A single fund (e.g. parent company, venture capital)
	Process	Set of variety of fund raising	Fund raising from market	A single investor

Table 6. Module Coupling Metrics

Coupling		Strong ⟷ Weak			
Modules		Control Coupling	Sequential Coupling	Common Coupling	Independent
People	Entity	Hierarchical Organization	Team with sequential processes	Team, Group	Independent organization
	Process	Direction, Order	Sequential processes	Information sharing	Independent governance
Things	Entity	Basic frame	Related parts	Common parts	Commercial (open) parts
	Process	Installation to basic frame	Sequential installation process	Shared supply in the group	Modularized process
Money	Entity	Capital governance	Conditional funds	Funds from market	Venture capital
	Process	Direct Management	Investment	Investment	Investment

Actor Transformation is most important among above operations. Examples of Actor Transformations are as follows:

(A) Automation:
- With machine automation
- With computer systems

(B) Handmade

(C) Improvement of internal structure:
- Improvement of production organization/process.
- Upgrading production machine for better performance.
- Upgrading computer system for better performance.

(D) Changing relationship with partners:
- Out-Sourcing
- In-Sourcing

3. INFORMATION SYSTEMS FRAMEWORK

Information Systems Framework has two transformations. One is Information Transformation (eTransformation), and the other is Software Structure Transformation (Software Transformation).

Information Transformation consists of following sequential procedures:

(1) Abstract information from *Modules.*

(2) Allocate information to Information System based on PIM.

Software Structure Transformation consists of a following procedure:

(3) Transform PIM to PSM (Platform Specific Model) with leveraging Component Framework and PSM Framework.

After these transformations, we will develop programs with the help of above PSM and Design Patterns.

3.1 Information Transformation (eTransformation)

3.1.1 Entity (Atom) & Information (Bit)

Every *Module* consists of physical Entity (Atom) and Information (Bit). Essentially in real world, Entity and Information are inseparable. However, digital technology and information technology have enabled the separation of information from Entity. Hereby, Information can be handled independently from the media and information processing efficiency has been largely improved.

On the other hand, because independent handling of Information from Entity has been enabled, it created many security problems such as

information without real Entity and synchronization error between Entity and Information.

By Actor Transformation (*Module* Substitution) Processes inside a *Module* are substituted. In this case, "Bit" is split from "Atom", and then, computer processes "Bit". After that "Bit" is synchronized with "Atom" to be re-integrated.

For the re-integration, MVC model that consists of Model, View and Controller is required.

Figure 4 shows MVC model.

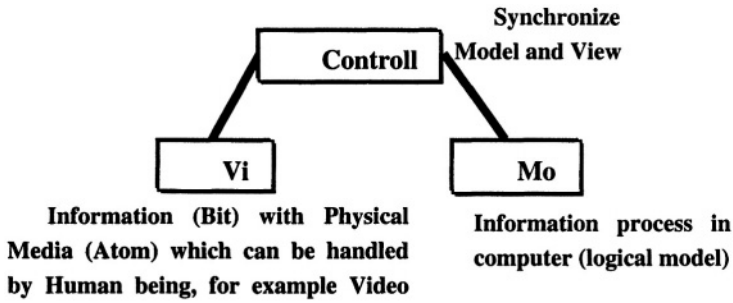


Figure 4. MVC Model

3.1.2 Procedure of Information Transformation

Information Transformation will be done according to the following hierarchies shown in Figure 6

(A) Splitting/Synchronizing Atom and Bit

Splitting Bit from Atom and synchronizing Atom and Bit is the most basic operation of Information Transformation. In case of order form as an example, picking out the described Bit such as product price and buyer's name means splitting Bit (i.e. order Information) from Atom (i.e. order form). MVC model is helpful to install this procedure into Information Systems.

(B) Information Processing

Information Processing is to process, accumulate and transform Bit that is picked out from Atom at above procedure. As to the order form, the typical example is to entry Bit that is picked out from order form into the database.

3-Dimensional Model of Information Processing as describe later is a good model that clarify the semantics of Information Processing. Appendix A illustrates the relation between Competitive Strategy, Module Transformation and the role of Information Processing.

(C) Knowledge Creation

Knowledge Creation is to create a new knowledge based on high-level Information Processing. The example of Knowledge Creation is to

analyze customer's purchase pattern by analyzing the customer data and purchase activity history.

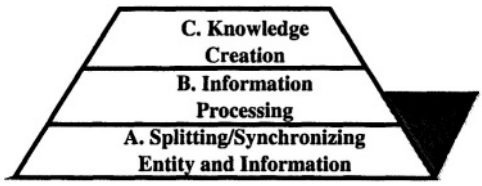


Figure 5. Hierarchy in the Role of Information

3.1.3 3-Dimensional Model of Information Processing

3-Dimensional Model of Information Processing that is used here is helpful to clarify the semantics of Information Processing. 3-Dimensional Model of Information Processing is shown in *Figure 6* consists of Horizontal (Common Use) Axis, Vertical (Value Chain) Axis and Time (Life Cycle) Axis.

Horizontal Axis means sharing variety of resources (physical resources and logical resources) via network. Not owning resources privately but sharing resources increases efficiency in the community.

Vertical (Value Chain) Axis means building value chains by connecting *Modules* and leveraging the value of each *Module.* It enables integration /coordination of skilled specialists worldwide, which increases efficiency in the global business operation.

Time (Lifecycle) Axis means supporting Module Lifecycles to upgrade quality with recording, storing, improving and reusing information.

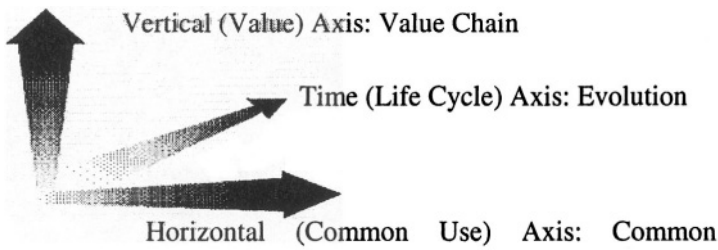


Figure 6. 3-Dimensional Model of Information Processing

3.2 Software Structure Transformation (Software Transformation)

3.2.1 Hierarchical Structure of IT System

Software system can be classified in 3 layers as shown in *Figure 7* – Basic Platform Layer, Managed Middleware Layer, and Application Layer.

Basic Platform Layer and Managed Middleware Layer belong to PSM (Platform Specific Model), and Application Layer belongs to PIM (Platform Independent Model). Basic Platform Layer and Managed Middleware Layer are often simply called "Platform", and Platform will be made of ready-made operating system, packaged software and software reusable components.

3.2.2 Software Structure Transformation in AP Layer

Software Structure Transformation in Application Layer consists of the following procedure:

(1) Build a MVC Model that contains Model, View and Controller. An example of MVC model is shown in *Figure 8* using cFramework structure, which is a product of EC-One, Inc. in Japan [8].
(2) Apply reusable *Module* in each element.

This methodology enables dispersed development module by module independently.

PSM described later is mapped from the PIM that is made hereby.

3.2.3 Software Structure Transformation in BP/MM Layer

The PSM that includes BP/MM Layer is built by transforming PIM, that is, PIM is mapped to PSM. The PSM is a combination of a variety of reusable components, which enables high-efficiency development.

Basic Platform Layer consists of variety of components, and many vendors release the products that fit each component. Appendix B illustrates examples of relation between BP Layer components and ready-made products. Appendix C illustrates examples of the detail of the BP Layer components in Sun Microsystems' Solaris environment.

Many vendors also release products that apply to MM Layer components as well as BP Layer components. Appendix D illustrates examples of relation between MM Layer components and ready-made products.

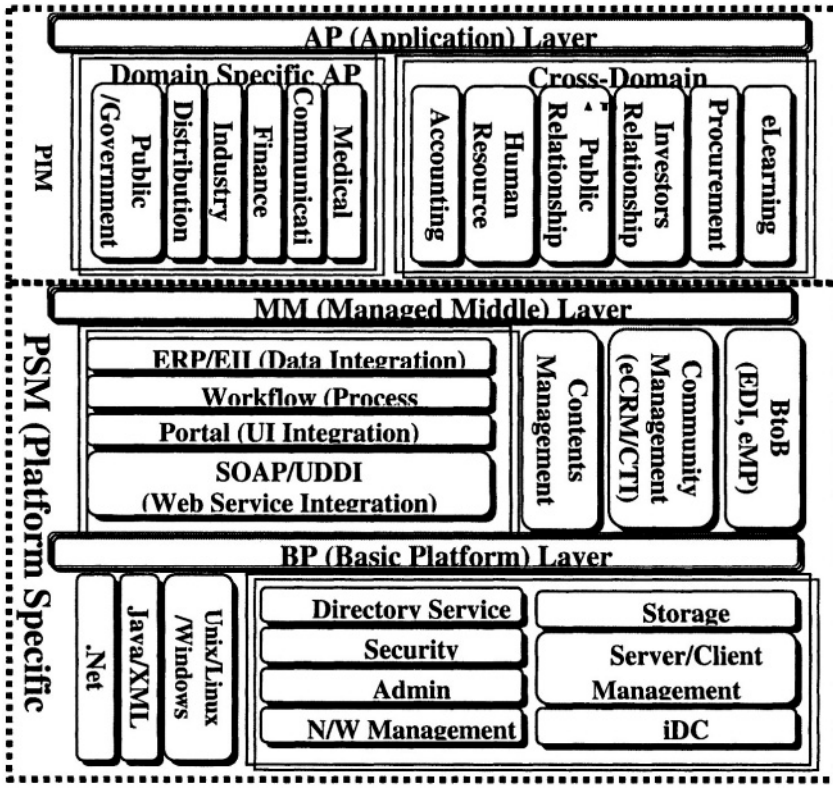


Figure 7. Hierarchical Structure of IT System

4. CONCLUSION

Globalization accelerates Business Disintegration and Modularization. We have discussed on "Enterprise Business Architecture Based on Modularization Concept". We also have presented a systematic methodology to map "Enterprise Business Architecture" to "Information Systems Framework". This mapping can be done as module-transformation, that is Business Module to PIM (Platform Independent Model), PIM to PSM (Platform Specific Model). The transformation is expected to bridge the gap between Business Systems Model and Information Systems Model.

We are enhancing UML (Unified Modeling Language) to describe Enterprise Business Architecture. With this enhancement, the transformation of modules is expected to have good support of automated tools.

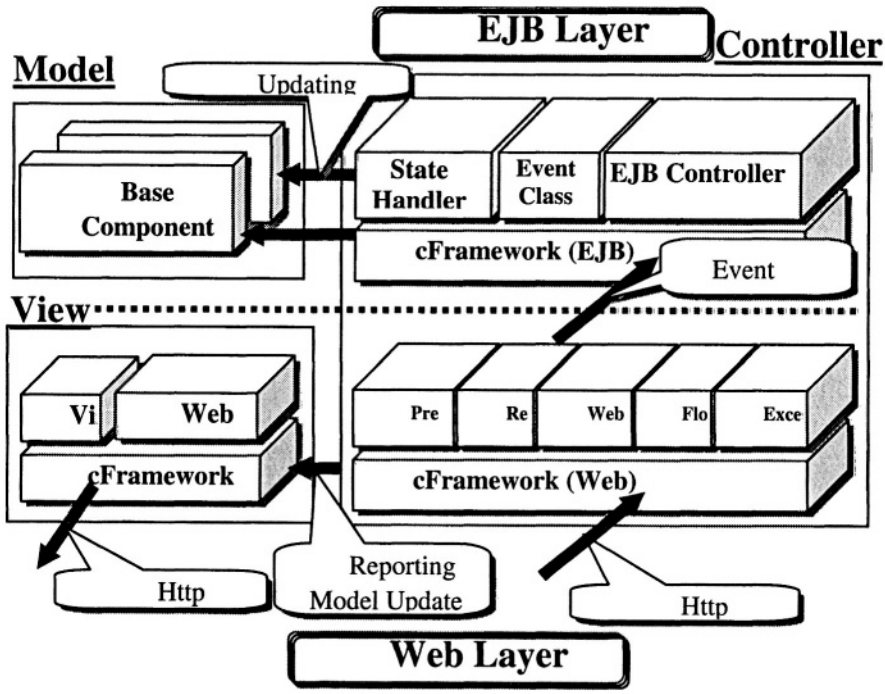


Figure 8. **Software Structure Transformation in Application Layer**

ACKNOWLEDGEMENT

The authors are extremely grateful to Mr. Koichi Yamano, CEO of Bunasoft Co. Ltd., who gave us a great deal of advices and Ms. Mika Sawada, who initiated this methodology.

REFERENCES

1. Aoki, Masahiko and Haruhiko Ando, *Modularity.* Tokyo: TOYO KEIZAI INC., 2002.
2. Myers, Glenford J, *Reliable Software through Composite Design.* Mason: Charter Publishers, Inc., 1975.
3. Eriksson, Hans-Erik and Magnus Penker, *Business Modeling With UML Business Patterns at Work: Business Patterns at Work,* OMG Press, Wiley Computer Publishing, 2000.
4. Mintzberg, Henry and Ludo Van der Heyden, *Organigraphs: Drawing How Companies Really Works,* Harvard Business Review, September-October, 1999.
5. Negroponte, Nicholas, *Being Digital,* New York:The Alfred A. Knopf, 1995.
6. The Object Management Group, http://www.omg.org/
7. The Object Technology Institute, Inc., http://www.otij.org/
8. EC-One, Inc., http://www.ec-one.com/

Appendix A: Extracting the role of "Information"

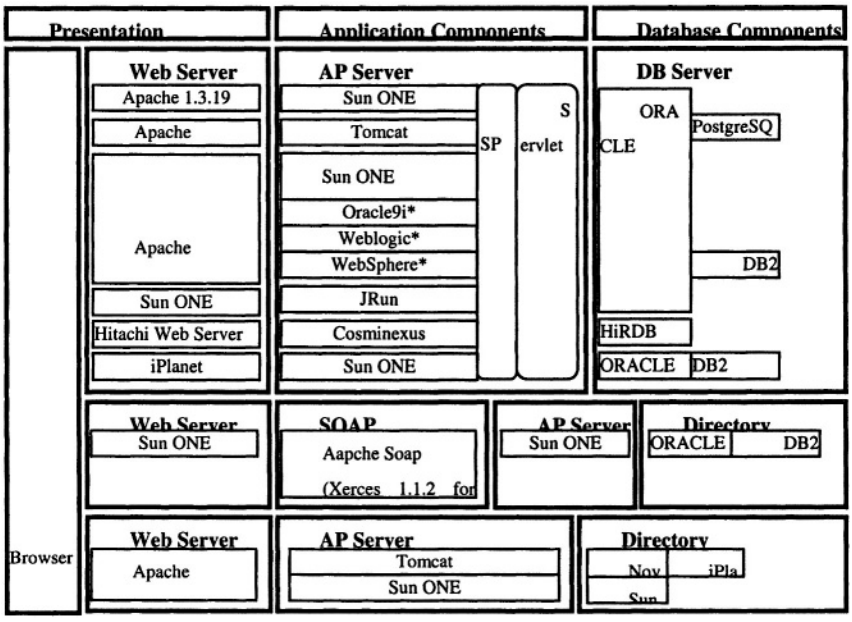
#	Competitive Strategy on Business System Model	Module (Actor) Transformation	Role of Information
A	Connecting Business Modules	Substituting the Interface	
A1	Computerizing Information Exchange between Modules	Computerizing Interface between Modules	EDI (Electronic Data Interchange)
A2	Business Modules related to the process from procuring parts to selling products are reorganized to form a "Value Chain"	Interface between Modules are computerized to form a "Value Chain"	SCM (Supply Chain Management)
B	Knowledge/Information Sharing among Business Modules	Formation of upper layer module consisting of Business Modules	
B1	Knowledge/Information sharing among Business Modules	sharing same knowledge /information	Web Portal
B2	Modules related to a specific business category are connected by workflow, enabling the circulation of electronic documents	Modules circulate electronic documents by workflow	Workflow
B3	Customer Information are shared among related Business Modules	Modules share the Customer Information and related processes	CRM (Customer Relationship Management)
C	Substituting Business Modules	Module Substitution (Actor Transformation)	
C1	Out-Sourcing	Business Module is substituted by Business Modules of other parties	EDI with Out-sourcing Partner
C2	Procure parts and office supply from eMP	Procurement partner is changed from current partner to eMP	eMP (e Market Place), MRO (Maintenance Repair and Order)
C3	Utilize Internet Service (e.g. ticket reservation, purchase)	Services provided by internal organization or by third party are substituted by Web Service	Web Service
C4	Distributed Manufacturing	Manufacturing Organi-zation Module is divi-ded and some portion of it is shifted to distri-buted manufac-turing facility.	EDI with distributed facility

Appendix B: Examples of PSM Models on BP Layer

Vendors / Compo-nents	Hitachi	IBM	Microsoft	Sun Micro systems	Oracle	Other
OS	HI-UX	AIX	Windows	Solaris		Linux
Web Browser			IE			NetScape Opera
Web Server	Hitachi Web Server	IBM HTTP Server	IIS	Sun ONE	Oracle9i AS*	Apache Zeus Novell exteNd .
RDB	HiRDB	DB2	SQL-Server		ORACLE	PostgreSQL
XML Parser		Web Sphere*	Biztalk	(MSXML)	Sun ONE AS*	XML Parser for Java WebLogic Server* (BEA) Xerces (Apache) The Expat XML Parser
Directory	Hitachi Directory Server		Microsoft	Sun ONE		Novell eDirectory
Security	Real Secure	Tivoli	ISA Server	Sun ONE	Oracle9i AS*	Novell iChain
Storage Mgt.	JP1/ Hi Command	Tivoli		Storage ONE		
System Administration	JP1	Tivoli	App. Center		Oracle9i AS*	

*Bundled with Server

Appendix C: PSM Layout for Solaris



Appendix D: Examples of PSM Modules on MM Layer

No	Vendors / Compo-nents	Hitachi	IBM	Microsoft	Sun Micro systems	Oracle	Novell	Other
1	Portal Server	Cosminexus	Web Sphere	Share Point Portal Server	Sun ONE	Oracle9i AS	Novell Portal Service	Web Logic (Ver7.0)
2	App. Servlet Engine	Cosminexus	Web Sphere	IE	Sun ONE	Oracle9i AS	Novell AS	Tomcat Web Logic JRun
3	SOAP/ UDDI			Microsoft SOAP Toolkit	Sun ONE		Novell Group Wise6	Apache SOAP
4	WF/ Group Ware	Groupmax	Notes/ Domino	SQL-Server				
5	ERP/EII					Oracle App.		SAP/R3P eople Soft
6	SCM/ CPFR	SCPLAN Logility Value Chain Solution						RHYTH MILOG
7	CTI/ CRM							Siebel People Soft 8
8	BtoB							
9	ECM		Content Mgr.	Content Mgt Server 2002		Oracle9i AS	Novell exteNd Composer	