



SEAMAIL

Newsletter from Software Engineers Association

Vol. 14, Number **2-3** May, 2004

目 次

Product Line Software Engineering	Kyo-Chul Kang	1
SOBA フォーラムの報告	編集部	37
ムンバイの暑い風	野村行憲	46
紺屋の白袴	SIGENV 有志	52
ISFST 2004 Call for Papers		59
編集後記		61

ソフトウェア技術者協会

Software Engineers Association

ソフトウェア技術者協会(SEA)は、ソフトウェアハウス、コンピュータメーカ、計算センタ、エンドユーザ、大学、研究所など、それぞれ異なる環境に置かれているソフトウェア技術者または研究者が、そうした社会組織の壁を越えて、各自の経験や技術を自由に交流しあうための「場」として、1985年12月に設立されました。

その主な活動は、機関誌 SEAMAIL の発行、支部および研究分科会の運営、セミナー/ワークショップ/シンポジウムなどのイベントの開催、および内外の関係諸団体との交流です。発足当初約200人にすぎなかった会員数もその後増加し、現在、北は北海道から南は沖縄まで、400余名を超えるメンバーを擁するにいたりました。法人賛助会員も20社を数えます。支部は、東京以外に、関西、横浜、名古屋、九州、広島、東北の各地区で設立されており、その他の地域でも設立準備をしています。分科会は、東京、関西、名古屋で、それぞれいくつかが活動しており、その他の支部でも、月例会やフォーラムが定期的に行われています。

「現在のソフトウェア界における最大の課題は、技術移転の促進である」といわれています。これまでわが国には、そのための適切な社会的メカニズムが欠けていたように思われます。SEAは、そうした欠落を補うべく、これからますます活発な活動を展開して行きたいと考えています。いまで日本にはなかったこの新しいプロフェッショナル・ソサイエティの発展のために、ぜひとも、あなたのお力を貸してください。

代表幹事： 荒木啓二郎

常任幹事： 熊谷章 高橋光裕 田中一夫 玉井哲雄 中野秀男 深瀬弘恭

幹事： 石川雅彦 大場充 落水浩一郎 窪田芳夫 小林修 小林允 桜井麻里
酒匂寛 塩谷和範 篠崎直二郎 新谷勝利 新森昭宏 杉田義明
中來田秀樹 野中哲 野村行憲 野呂昌満 端山毅 平尾一浩
藤野誠治 松原友夫 渡邊雄一

事務局長： 岸田孝一

会計監事： 橋本勝 吉村成弘

分科会世話人 環境分科会(SIGENV)：塩谷和範 田中慎一郎 渡邊雄一
教育分科会(SIGEDU)：君島浩 篠崎直二郎 杉田義明 中園順三
ネットワーク分科会(SIGNET)：人見庸 松本理恵
プロセス分科会(SEA-SPIN)：伊藤昌夫 塩谷和範 高橋光裕 田中一夫 端山毅 藤野誠治
フォーマルメソッド分科会(SIGFM)：荒木啓二郎 伊藤昌夫 熊谷章 佐原伸 張漢明 山崎利治
オープンソース分科会(SIGOSS)：石川雅彦 岸田孝一 杉田義明 鈴木裕信 中野秀男

支部世話人 関西支部：小林修 中野秀男 横山博司
横浜支部：野中哲 藤野見延 北條正顕
名古屋支部：石川雅彦 角谷裕司 野呂昌満
九州支部：杉田義明 武田淳男 平尾一浩
広島支部：大場充 佐藤康臣 谷純一郎
東北支部：布川博士 野村行憲

賛助会員会社：ジェーエムエーシステムズ SRA PFU テブコシステムズ 富士通
オムロンソフトウェア キヤノン 新日鉄ソリューションズ
ダイキン工業 オムロン 富士電機 プラザー工業 オリンパス
リコー NTTデータ ヤマハ オープンテクノロジーズ
SRA西日本 日本総合研究所 SRA東北
(以上20社)

SEAMAIL Vol. 14, No. 2-3 2004年5月28日発行 編集人 岸田孝一
発行人 ソフトウェア技術者協会(SEA)
〒160-0004 東京都新宿区四谷3-12 丸正ビル5F
T: 03-3356-1077 F: 03-3356-1072 E-mail: sea@sea.or.jp URL: http://www.sea.jp
印刷所 有限会社 錦正社 〒130-0013 東京都墨田区錦糸町4-3-14
定価 1,000円 (禁無断転載)



Pohang University of Science and Technology
(POSTECH)

Software Engineering Laboratory

Product Line Software Engineering:
Concepts, Efforts, and Issues

Kyo C. Kang

Software Engineers Association

April 6th 2004



Pohang University of Science and Technology (POSTECH)



Copyright © 2004 SE Lab. Dept. of CSE
POSTECH, R.O. Korea

Agenda

- Introduction
- Efforts for Product Line Engineering
- Feature-Oriented Reuse Method
- Other Issues

POSTECH
Copyright © 2004
SE Lab. Dept. of CSE
POSTECH, R.O. Korea



1

Title : Product Line Software Engineering:
Concepts, Efforts, and Issues

© 2004 by Pohang University of Science and Technology
Department of Computer Science and Engineering
Software Engineering Laboratory, R.O. Korea



What is a Software Product Line ?

Introduction

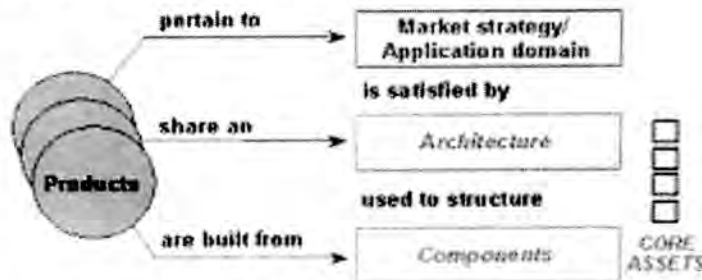
■ A software product line is a *set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.*

What is a Software Product Line ?

Introduction



Software Product Lines



Product lines
• take economic advantage of commonality
• bound variability



What is a Software Product Line ?

Introduction

Software Engineering Institute

The Key Concepts

Use of a common asset base

in production

of a related set of products



Architecture

Production Plan

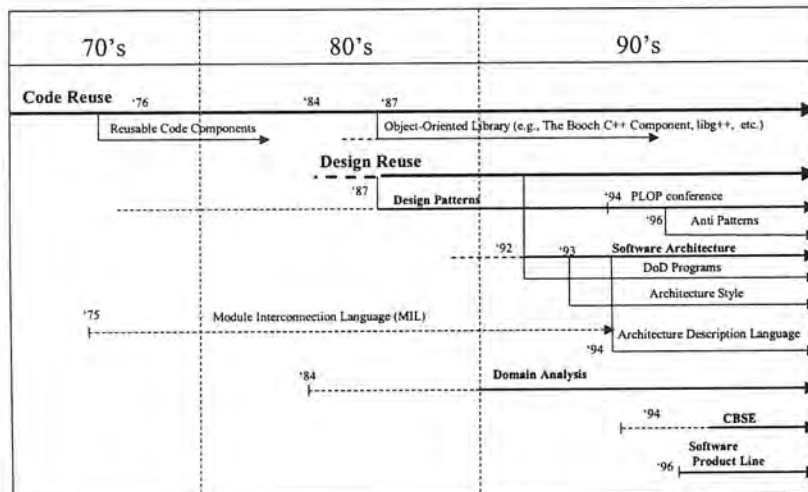
Scope Definition Business Case

POSTECH Copyright © 2004 SE Lab, Dept. of CSE POSTECH, R.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>

Software Reuse Technology Evolution

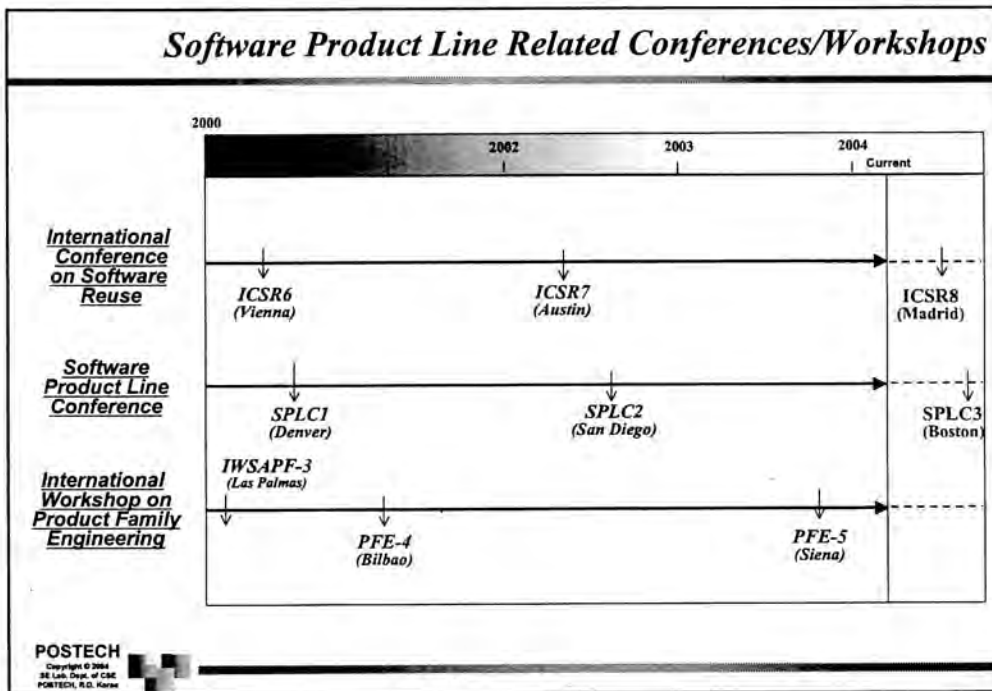
Introduction



POSTECH Copyright © 2004 SE Lab, Dept. of CSE POSTECH, R.O. Korea



Software Product Line Related Conferences/Workshops



SEI: Software Product Line Practice





Fraunhofer IESE: PuLSE

Fraunhofer IESE
Institut Experimentelles Software Engineering

PuLSE : Product Line Software Engineering

The diagram illustrates the PuLSE architecture. On the left, 'Specification Models' include System, Component, and Software models. In the center, 'Deployment Phases' show a flow from 'Software Architecture' to 'Implementation' and 'Deployment'. On the right, 'Technical Components' include 'Software Architecture', 'Software Platform', 'Software Library', and 'Software Support'. Below these are 'Support Components' like 'Product Line Tools', 'Product Line Process', and 'Product Line Support'.

KobrA: Komponent based application development

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea

http://www.iese.fhg.de/Products_Services/

Activities in Europe

The timeline shows the progression of European activities. On the left, 'ARES (1995-1999)' and 'PRAISE (1998-2000)' are grouped under the 'Esprit Project (1994-1998)'. In the center, 'ESAPS (1999-2001)' is shown with a handwritten 'ESAPS' below it. To its right, 'CAFE (2001-2003)' is shown with a small car icon. On the far right, 'FAMILIES (2003-2005)' is shown with a small globe icon. These three (ESAPS, CAFE, FAMILIES) are grouped under the 'ITEA Project (1999-)'.

ARES (1995 - 1999)

PRAISE (1998 - 2000)

ESAPS (1999 - 2001)

CAFE (2001 - 2003)

FAMILIES (2003 - 2005)

Esprit Project (1994 - 1998)

ITEA Project (1999 -)

ARES : Architectural Reasoning for Embedded Systems
PRAISE : Product-line Realization and Assessment in Industrial Settings
ESAPS : Engineering Software Architectures, Processes, and Platforms for System families
CAFE : Concepts to Application in System-Family Engineering
FAMILIES : FAct-based Maturity through Institutionalization Lessons-learned and Involved Exploration of System-family engineering
Esprit : The EU information technologies programme
ITEA : Information Technology for European Advancement

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea



Esprit project

- Esprit
 - an integrated programme of industrial R&D projects and technology take-up measures
 - a part of the EU's Fourth Framework Programme
- Framework programmes
 - Purpose
 - to set the overall aims and priorities for the EU's R&D activities
 - to define the content of the work in broad terms
 - Duration : 1994 – 1998
 - Area
 - Information and Communication Technologies
 - Industrial Technologies
 - Environment
 - Life Sciences and Technologies
 - Energy
 - Research & Training in the Sector
 - Transport
 - Targeted Socio-Economic Research
 - etc..

<http://www.cordis.lu/en/home.html>
<http://www.cordis.lu/esprit/src/intro.htm>
<http://www.cordis.lu/esprit/home.html>

POSTECH

Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea



ITEA project

- ITEA : Information Technology for European Advancement
- ITEA is an eight-year strategic pan-European programme for advanced pre-competitive research and development in embedded and distributed software
- ITEA was set up to stimulate and support projects that will give European industry a leading edge in software-intensive systems. We work closely with other EUREKA projects and the Framework Programmes of the European Commission.
- Funding : all 33 countries in the EUREKA framework
 - EUREKA is a pan-European network for market-oriented, industrial R&D.

<http://www.eureka.be>
<http://www.itea-office.org/>

POSTECH

Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea





ARES

- ARES : Architectural Reasoning for Embedded Systems
- Objective
 - To enable software developers to explicitly describe, assess, and manage architectures of embedded software families
- Project number : Esprit project 20477
- Partner
 - Companies : Nokia (prime contractor), Philips, ABB
 - Colleges : Imperial College, Technical University of Vienna
Technical University of Madrid
- Start Date : 01-Dec-1995
- End Date : 28-Feb-1999
- Duration : 39 months
- Program Acronym : ESPRIT 4
- Subprogram Area : Software Intensive Systems Engineering
- Program Type : 4th FWP (Fourth Framework Program)

<http://www.infosys.tuwien.ac.at/staff/hg/projects/ARES/>

<http://dbs.cordis.lu/cordis-cgi/showsrch?CALLER=PROJESPRITADVANCEDSRCH>

POSTECH

Copyright © 2004
36 Lab. Dept. of CSE
POSTECH, P.O. Korea



PRAISE

- PRAISE : Product-line Realization and Assessment in Industrial Settings
- Objective
 - To provide integrated and validated methodological support to the development of large software product families in industrial settings
- Project number : Esprit project 28651
- Partner
 - Thomson-CSF LCR (France) – Prime Contractor
 - Ericsson Software Technology (Sweden)
 - ESI(European Software Institute)
 - BOSCH (Germany)
- Start Date : 01-Sep-1998
- End Date : 29-Feb-2000
- Duration : 18 months
- Program Acronym : ESPRIT 4
- Subprogram Area : Software Intensive Systems Engineering
- Program Type : 4th FWP (Fourth Framework Program)

<http://www.esi.es/en/Projects/Praise/praiseProject.html>

<http://dbs.cordis.lu/cordis-cgi/showsrch?CALLER=PROJESPRITADVANCEDSRCH>

POSTECH

Copyright © 2004
36 Lab. Dept. of CSE
POSTECH, P.O. Korea





ESAPS Project

- ESAPS : Engineering Software Architectures, Processes, and Platforms for System families
- Objective
 - To provide an enhanced system-family approach by combining the most promising technologies of the partner-companies in the following areas: Analysis, Definition and Evolution of system-families.



Project ID : ITEA 99005
Start Date : July 1999
End Date : June 2001

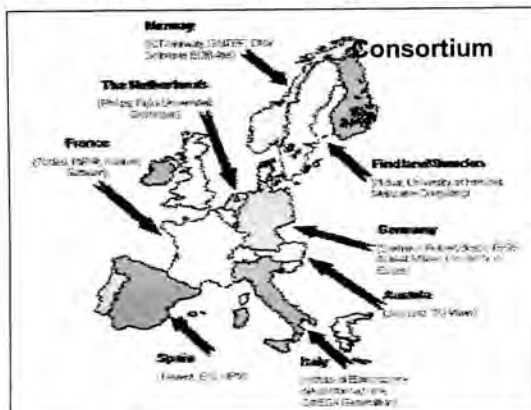
ESI (European Software Institute) has a special role in the project as a means for distributing the experience to other European companies

POSTECH
Copyright © 2004
All Rights Reserved
POSTECH, P.O. Korea

<http://www.esi.es/en/Projects/esaps/esaps.html>

CAFÉ Project

- CAFÉ : Concepts to Application in System-Family Engineering
- Objective
 - To extend the results of the ESAPS by providing methods and processes that support the whole life cycle of system families



Project ID : ITEA 00004
Start Date : July 2001
End Date : June 2003

POSTECH
Copyright © 2004
All Rights Reserved
POSTECH, P.O. Korea

<http://www.esi.es/en/Projects/Cafe/cafe.html>



FAMILIES Project

- FAMILIES : FACT-based Maturity through Institutionalization Lessons-learned and Involved Exploration of System-family engineering
- Objective
 - To finalize two earlier ITEA project (ESAPS and CAFÉ)



Project ID : ITEA 02009
Start Date : July 2003
End Date : June 2005

POSTECH

Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea

<http://www.esl.es/en/Projects/Cafe/cafe.html>

Celsius Tech



Carlisle Mellon
Software Engineering Institute

CelsiusTech: Ship System 2000

A family of 55 ship systems

integration test of 1.1.5 million
SLOC requires 1.2 people
rehosting to a new platform/OS
takes 3 months
cost and schedule targets are
predictably met
performance/distribution behavior
known in advance
customer satisfaction is high
hardware-to-software cost ratio
changed from 35:65 to 60:20



POSTECH

Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>



Cummins Inc.

Carnegie Mellon
Software Engineering Institute

Cummins Inc.: Diesel Engine Control Systems

Over 20 product groups with
over 1000 separate engine
applications

product cycle time was
slashed from 250 person-
months to a few person
months

Build and integration time was
reduced from one year to one
week

quality goals are exceeded
customer satisfaction is high
product schedules are met



POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, P.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>

Raytheon

Carnegie Mellon
Software Engineering Institute

National Reconnaissance Office / Raytheon: Control Channel Toolkit

Ground-based spacecraft
command and control systems

increased quality by 10X
incremental build time
reduced from months to
weeks

software productivity
increased by 7X
development time and costs
decreased by 50%
decreased product risk



POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, P.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>



Market Maker GmbH



Emerging Market
Software Engineering Institute

Market Maker GmbH: MERGER

Internet-based stock
market software

each product "uniquely"
configured
three days to put up a
customized system



POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, P.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>

Nokia



Emerging Market
Software Engineering Institute

Nokia Mobile Phones

Product lines with 25-30 new
products per year

- Across products there are
- varying number of keys
 - varying display sizes
 - varying sets of features
 - 58 languages supported
 - 130 countries served
 - multiple protocols
 - needs for backwards compatibility
 - configurable features
 - needs for product behavior change after release



POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, P.O. Korea

<http://www.sei.cmu.edu/plp/essentials/>



Phillips

TV

Medical Systems

X-Ray

Ultrasound
Surgery

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea

The Koala Component Model

Koala offers:

- 1) Provides interfaces and instantiations as first class citizens
- 2) Manages interfaces and justifies bindings
- 3) Aggregates and Glazes
- 4) Parameterizations, optional interfaces and Dynamic Binding

14

Robert Bosch GmbH

Car Periphery Supervision

Feature Modelling

Architecture Modelling

Architecture Modelling

Steffen Thiel and Andreas Hein, Modeling and Using Product Line Variability in Automotive Systems, IEEE Software, July/August 2002, pp.66-72

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea



Feature-Oriented Product Line Method

FORM

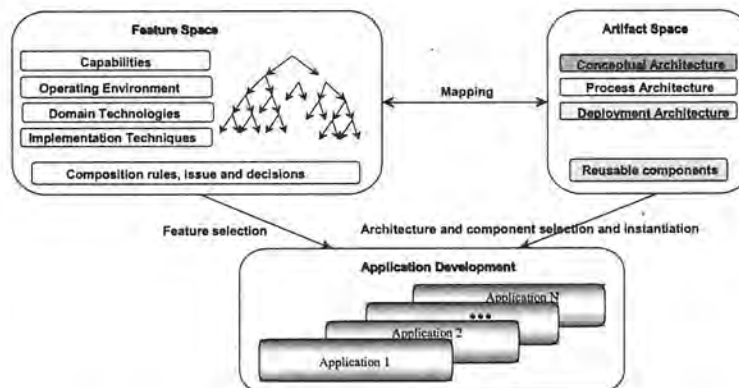
■ FORM: Feature-Oriented Reuse Method with Product Line Specific Reference Architectures

- A model-based product line method
- Focuses on architecture and component reuse
- Extended from the CMU SEI's FODA method
- ***Uses a marketing and product plan as a key driver***



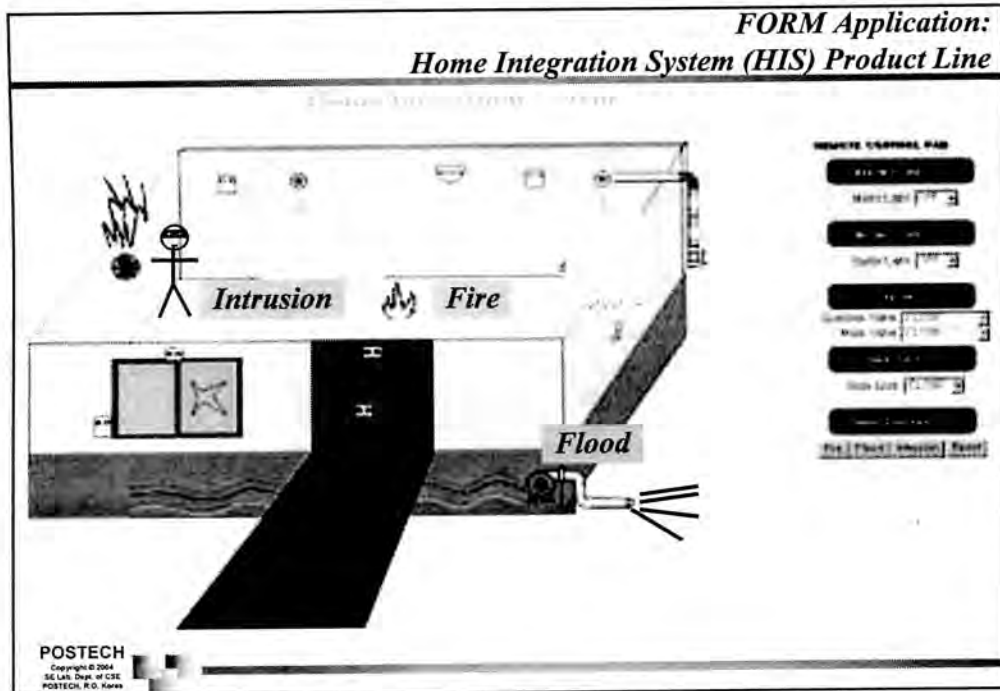
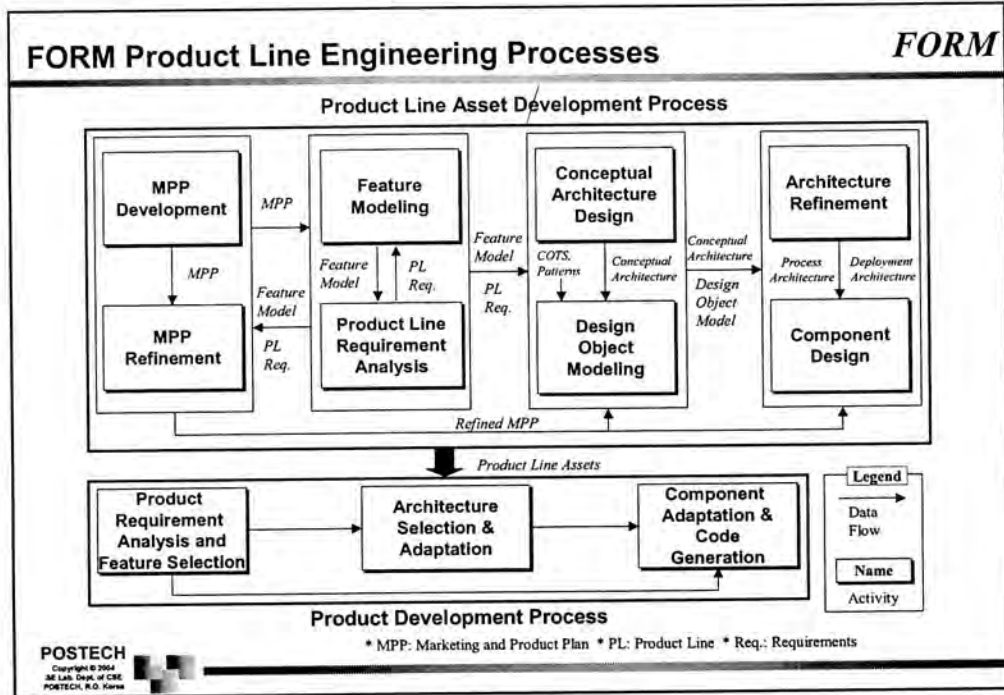
Method Concept

FORM



The core of FORM lies in the analysis of domain features and the use of these features to develop reusable and adaptable domain artifacts.







Marketing and Product Plan (MPP)

Product Line Initiation

What features will be included in the products?

How they will be delivered?

- Pre-packaged product
- Customer selectable from a set of features
- Custom-made products

What extensions are allowed?
How they can be extended?

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea

Marketing and Product Plan (MPP)

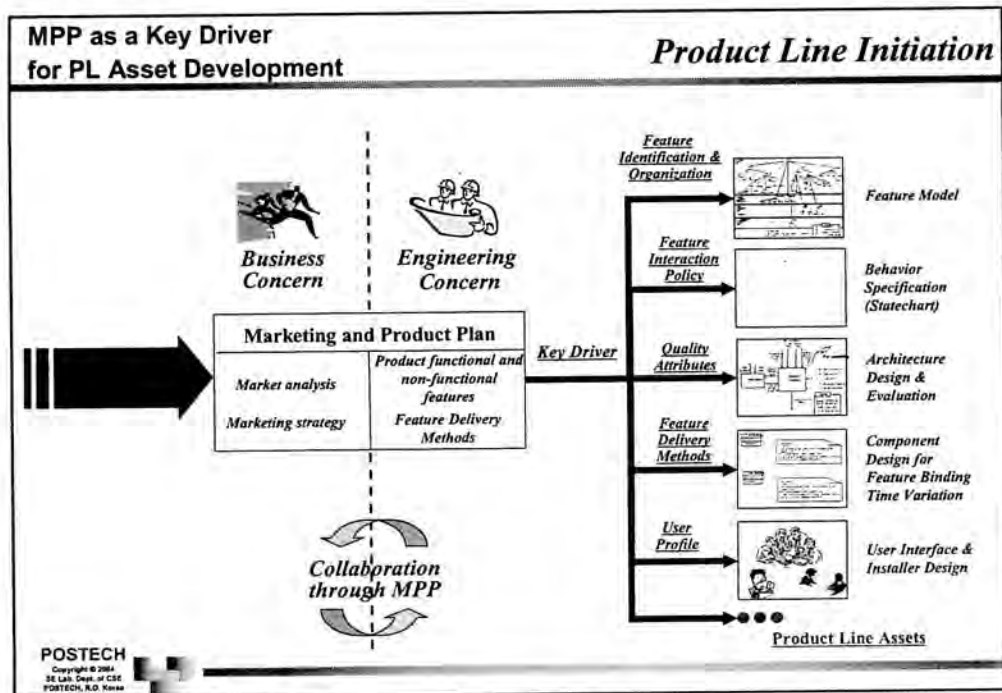
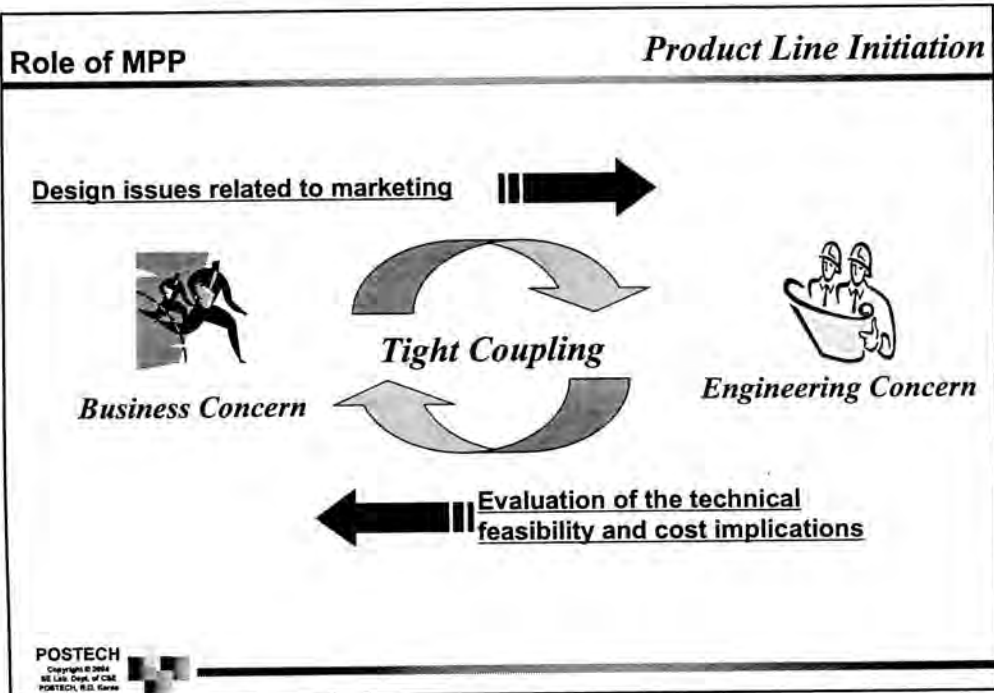
Product Line Initiation

Different cultural traits and legal constraints

Different computing knowledge backgrounds

Different computing environments

POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, R.O. Korea





Elements of MPP

Product Line Initiation



POSTECH
Copyright © 2004
36, Link 1st, of CSE
POSTECH, P.O. Korea

MPP for HIS PL

Product Line Initiation

Marketing and Product Plan for HIS PL		
Market Segments	Office building: High-End (HE) product	Household: Low-End (LE) product
User/Maintainer Profile	Dedicated engineers with computer science backgrounds.	No computer knowledge is assumed.
Legal Constraints	Emergency control services must conform to codes of each country.	Emergency control services must conform to codes of each country.
Feature Delivery Method	Feature selection from a predefined set of features (Feature Selection Method)	Prepackaged Method
Product Features	Fire, Intrusion, Flood, Security, and other customer specific features	Fire, Intrusion, Flood
Quality Attributes	Safety, Reliability, Scalability	Safety, Reliability, Scalability, Usability
Product Feature Binding Time	Product Delivery Time	Product Build Time

POSTECH
Copyright © 2004
36, Link 1st, of CSE
POSTECH, P.O. Korea



What is a feature?

Product Line Analysis

- Understanding *Commonalities and Variabilities* is the key to successful Product Line Software Engineering.
- FORM analyzes the commonalities and variabilities in terms of *features*.

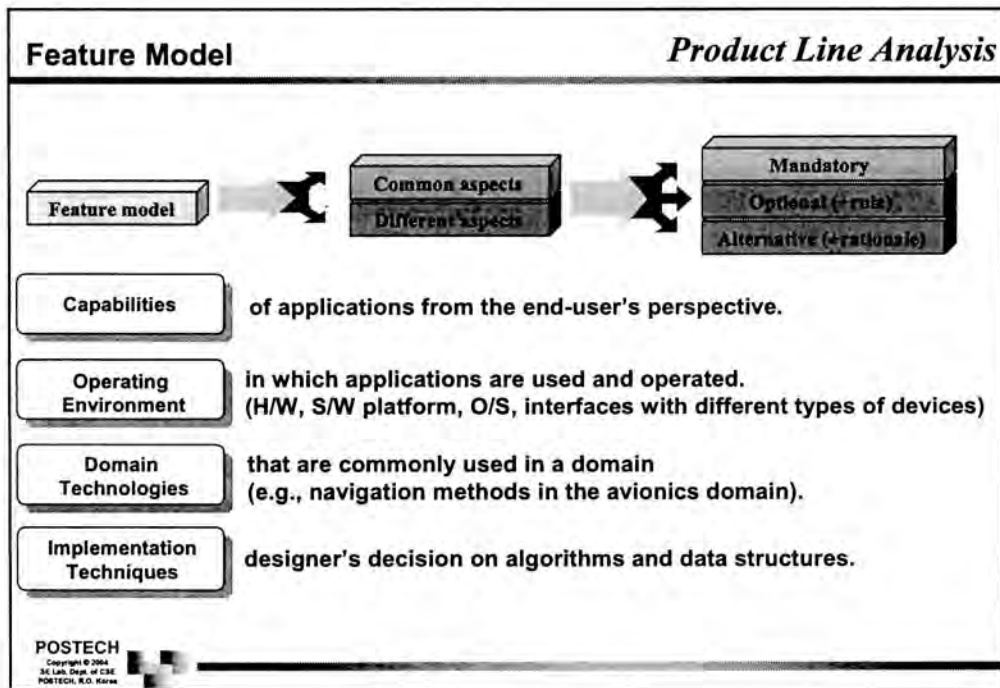
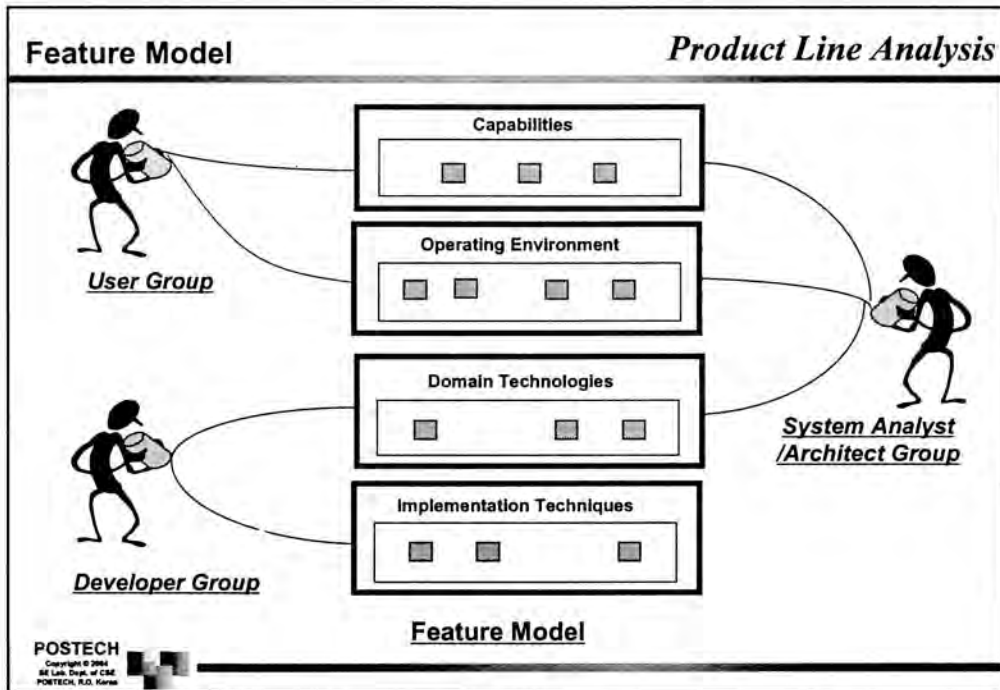


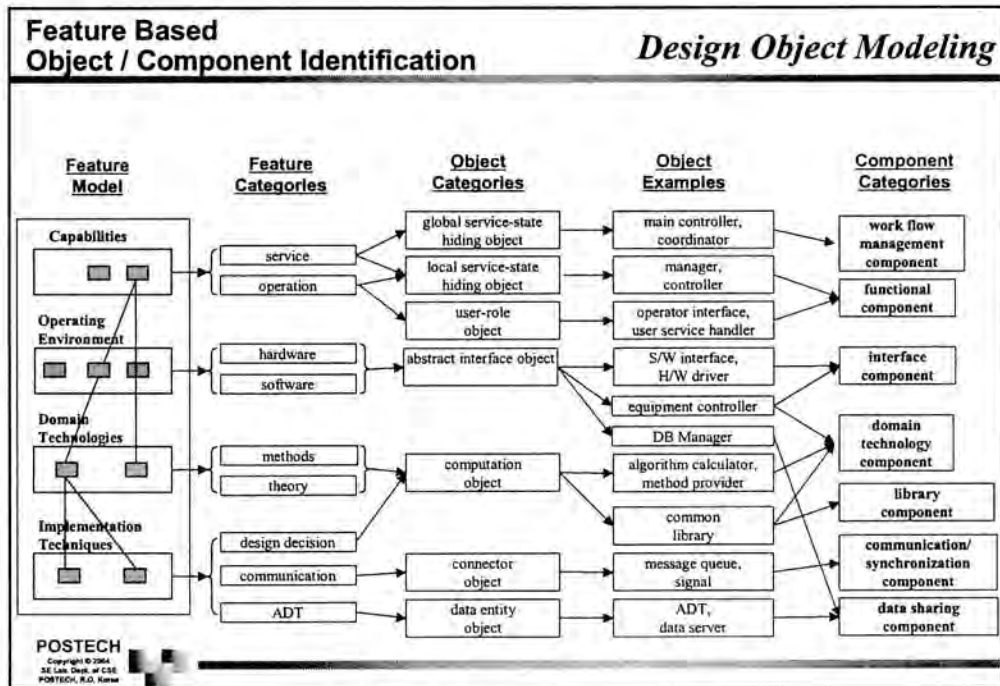
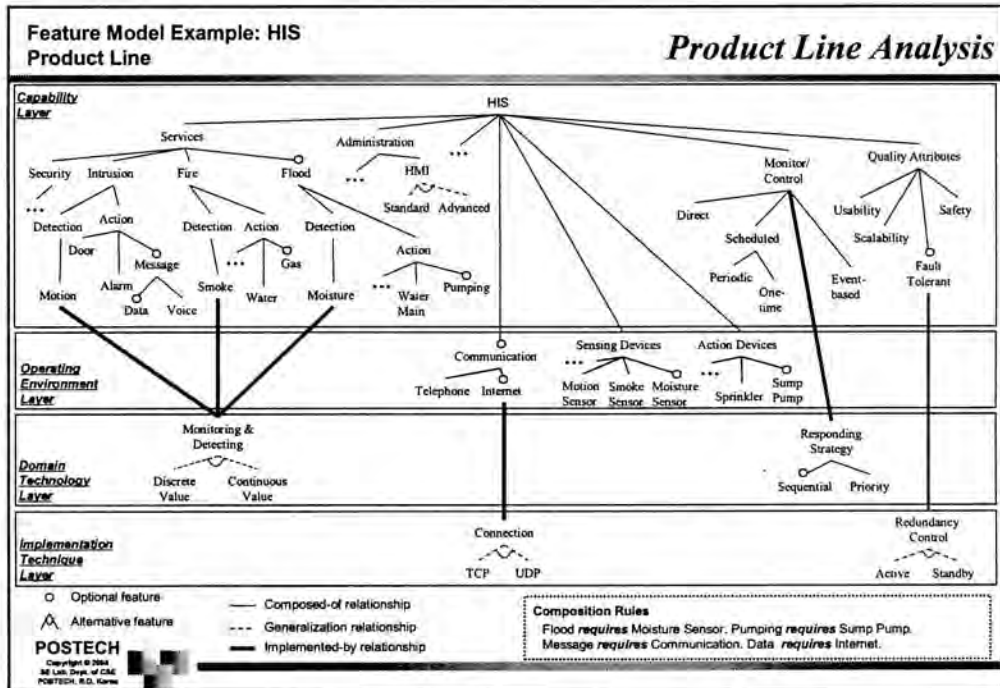
What is feature?

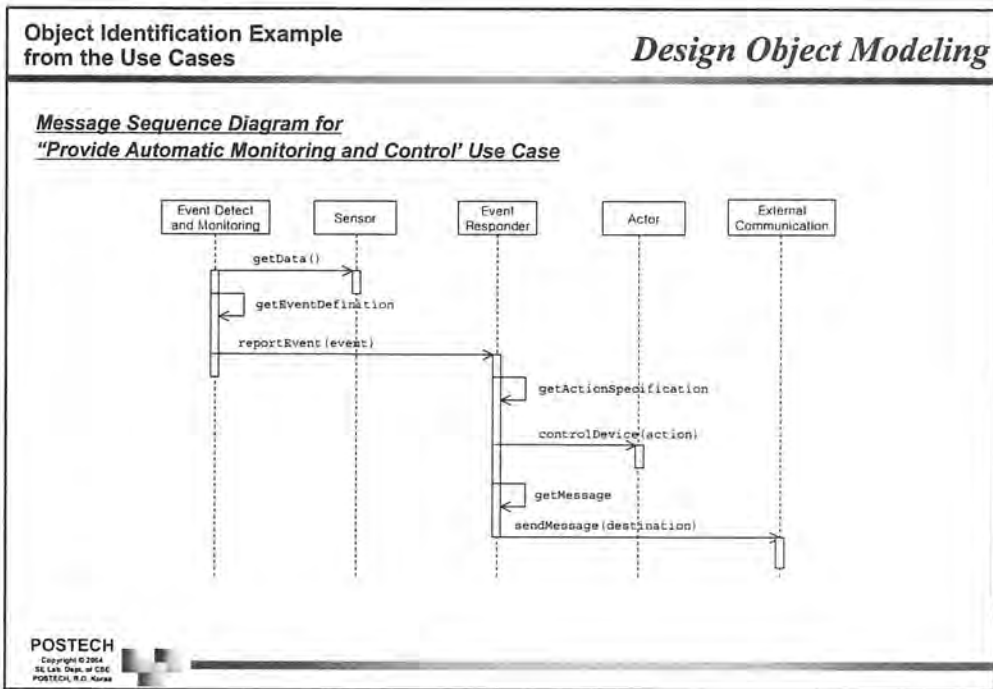
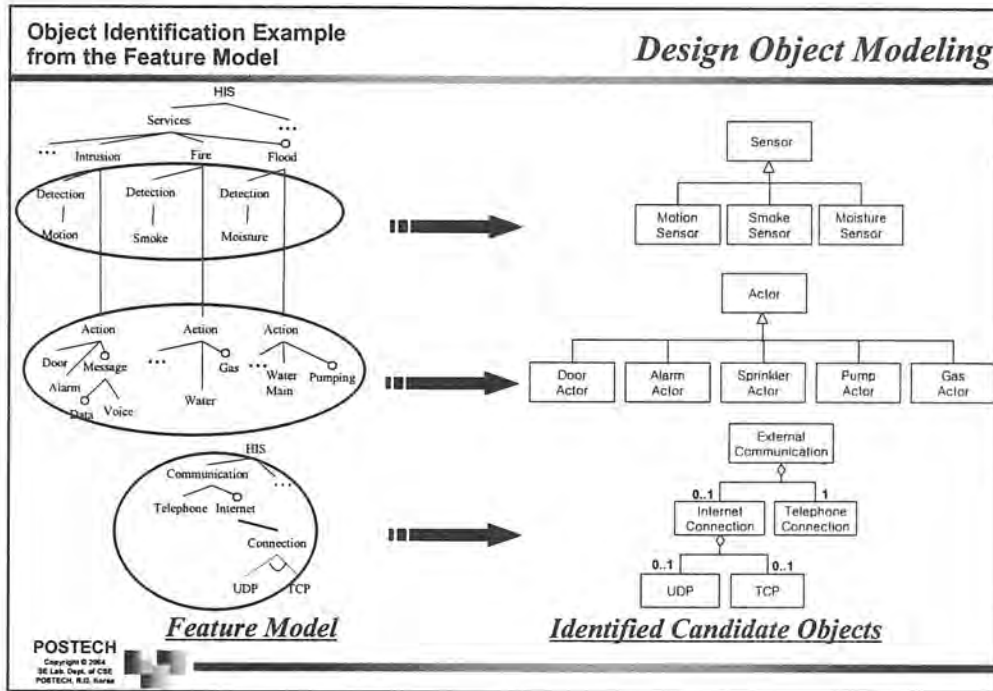
Product Line Analysis

- “An abstraction of any prominent and distinctive concepts or characteristics that are visible to various stakeholders”
- Example: Call_Answer, Call_Request, Call_Forwarding, etc.





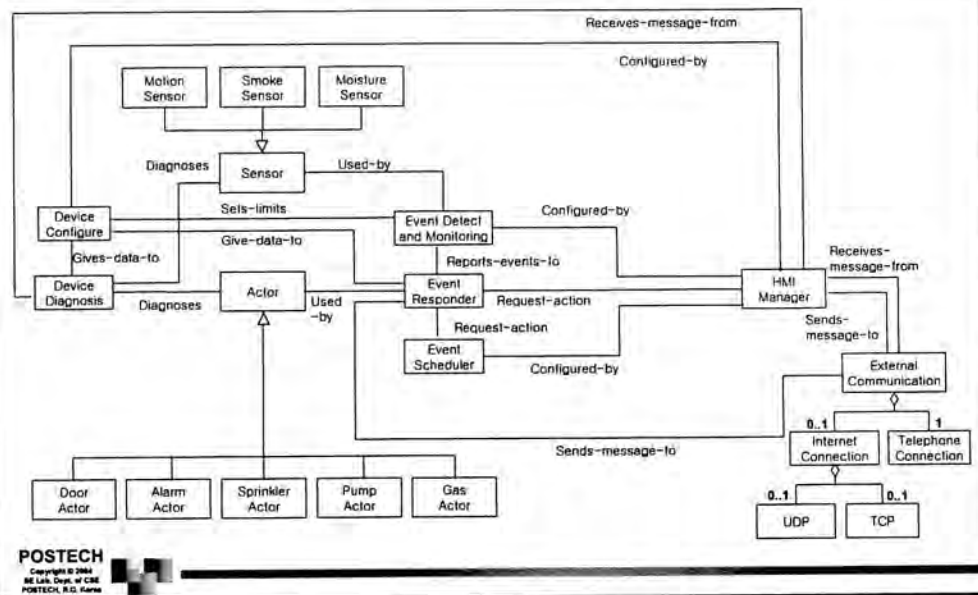






Design Object Model Example

Design Object Modeling

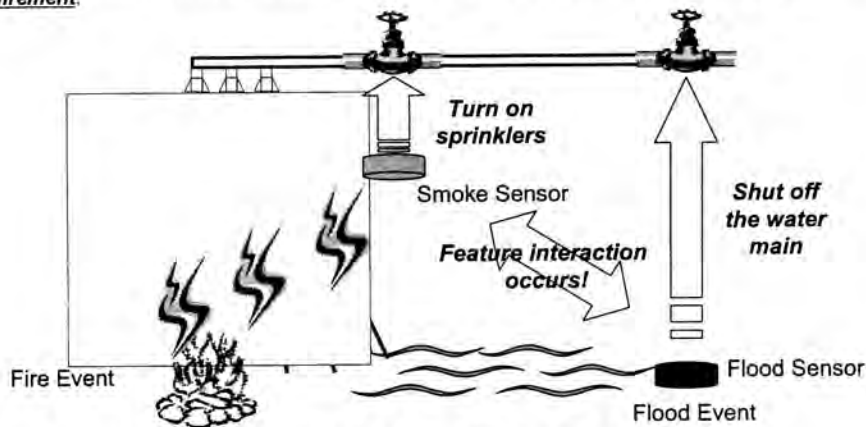


POSTECH
Copyright © 2004
All Rights Reserved

Feature Interaction Problem

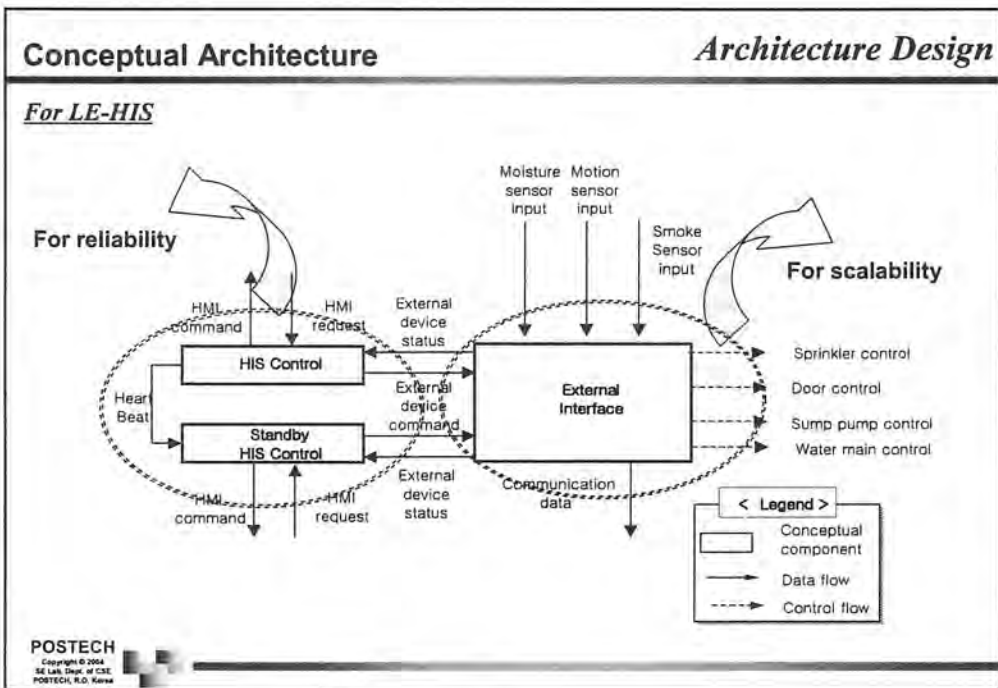
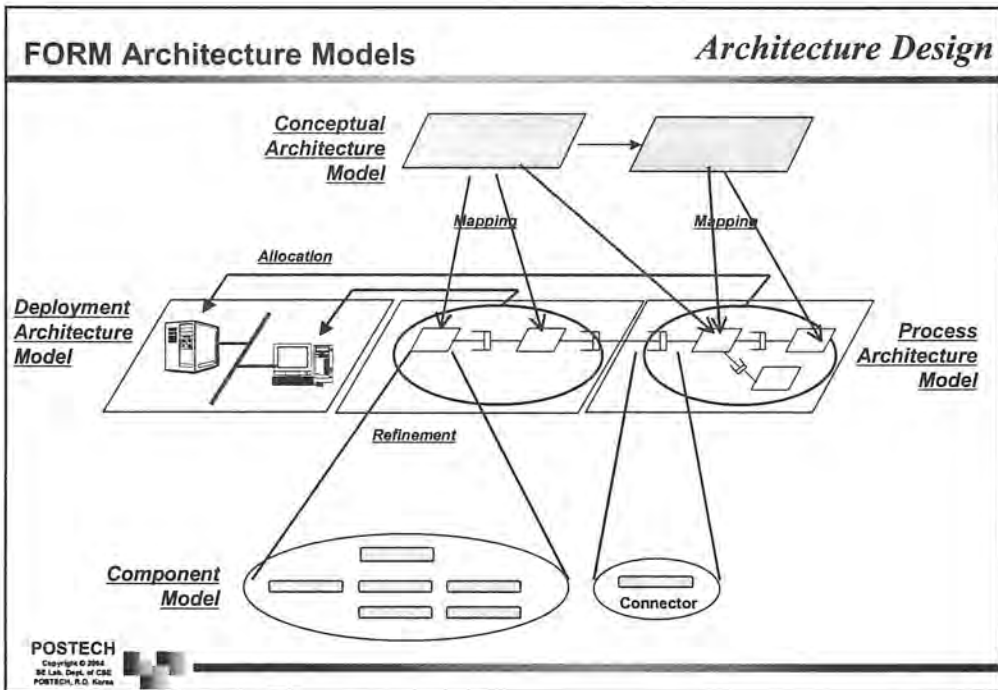
Feature Interaction Analysis

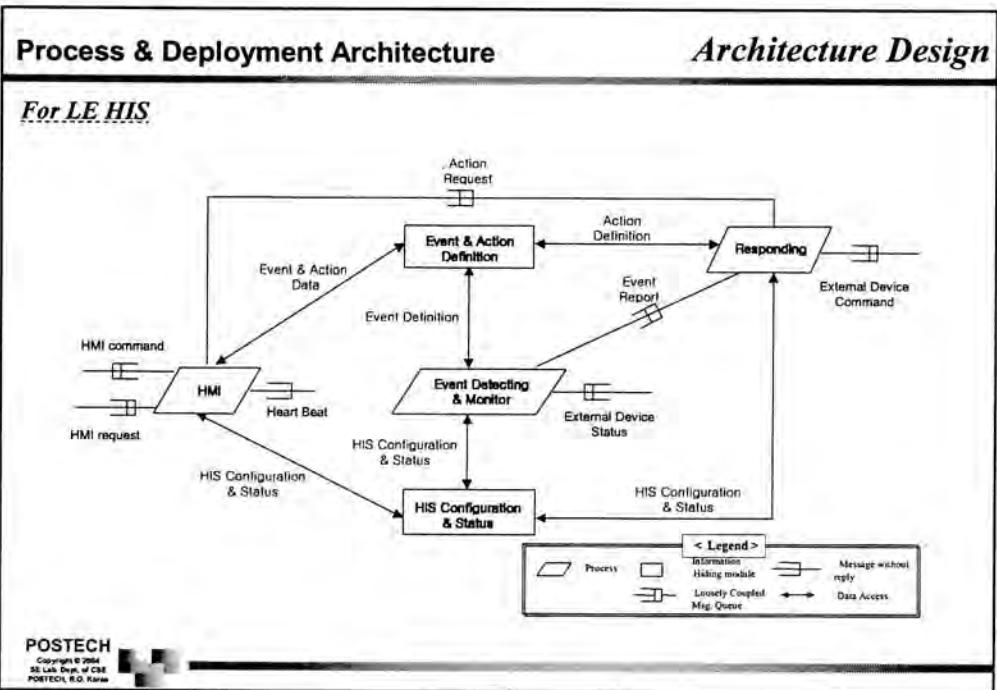
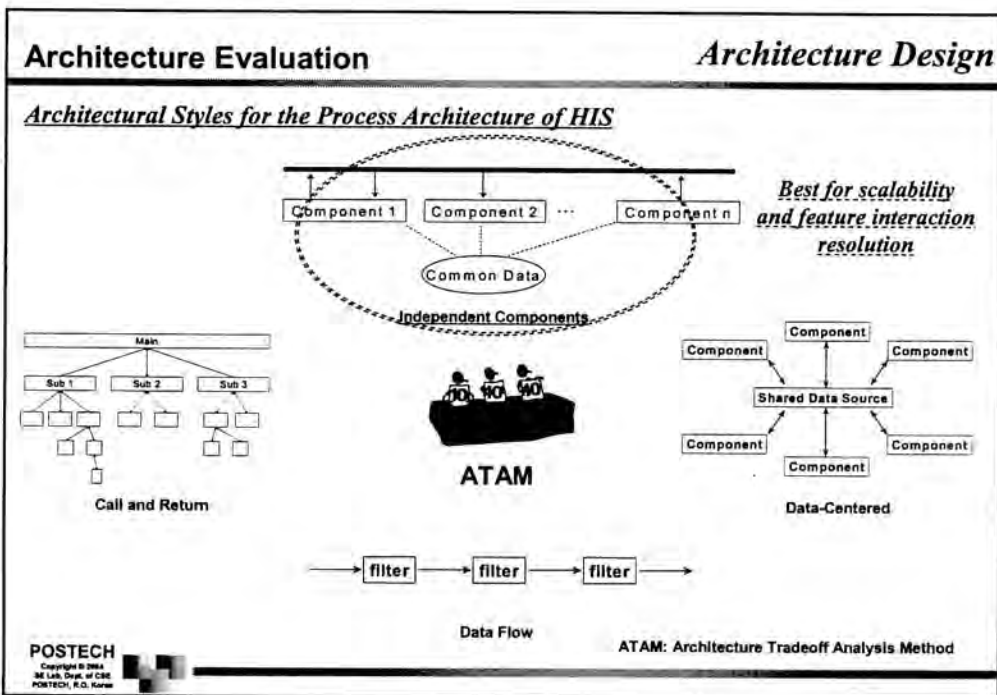
The problem of unexpected side-effects when a feature is added to a set of features is generally known as the feature interaction problem. It should be carefully analyzed and resolved to meet the **safety requirement**.



Feature interaction between the fire and flood features

POSTECH
Copyright © 2004
All Rights Reserved



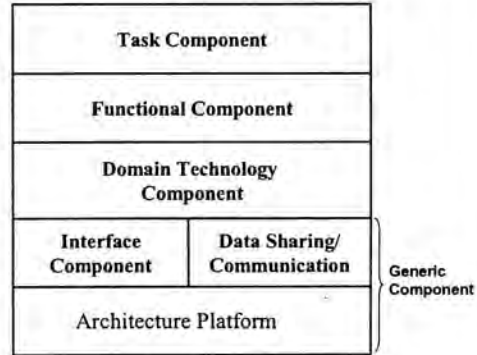




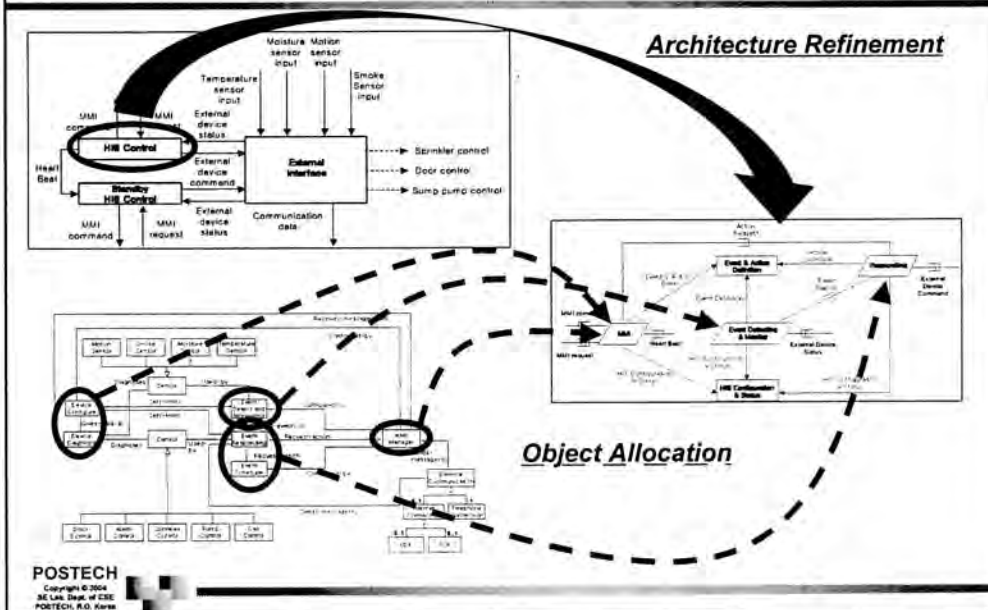
Component Types

Architecture Design

- Application Task Components
 - Control and coordination
- Functional Components
 - Services/operations
- Domain Technology Components
 - Algorithms/techniques that are commonly used in the domain
- Data Communication Components
 - Message queue, task synchronization, etc.
- (Environment) Interface Components
 - Device driver, user interface, etc.
- Architecture Platform
 - Unix pipe, CORBA, RPC



Allocation of Objects to Architecture Models





FORM Component Specification

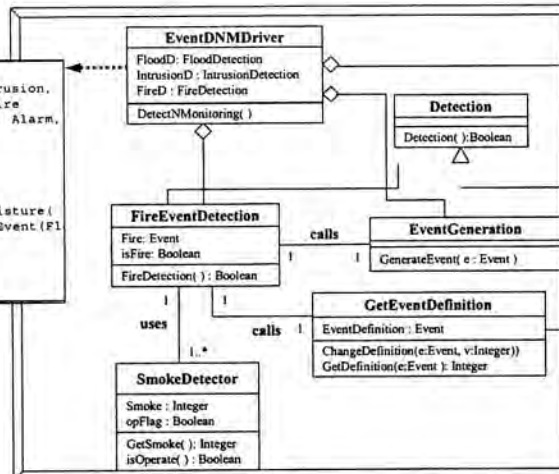
Component Design

FORM component specification for macro processing

```

Component
  EventDNMDriver; $Flood, $Intrusion,
  Implements Flood, Intrusion, Fire
  Require Moisture, SumpPump, Alarm,
{
  DetectNMonitoring() {
    $IF($Flood) {
      If ( (moistureSensor->GetMoisture()
        eventGeneration->GenerateEvent(F1)
      )
    }
  }
}

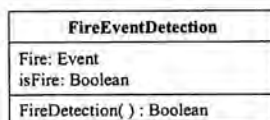
```



OCL Specification

Component Design

- This component detects a fire event.
 - If the smoke level from smoke detectors is bigger than the smoke value defined in the event definition, this component generates a fire event.



```

FireEventDetection::FireDetection()
pre:
smokeDetector.opFlag = true
post:
if smokeDetector->GetSmoke() >=
  getEventDefinition->GetDefinition(Fire)
then self.isFire = true
else self.isFire = false
endif

```



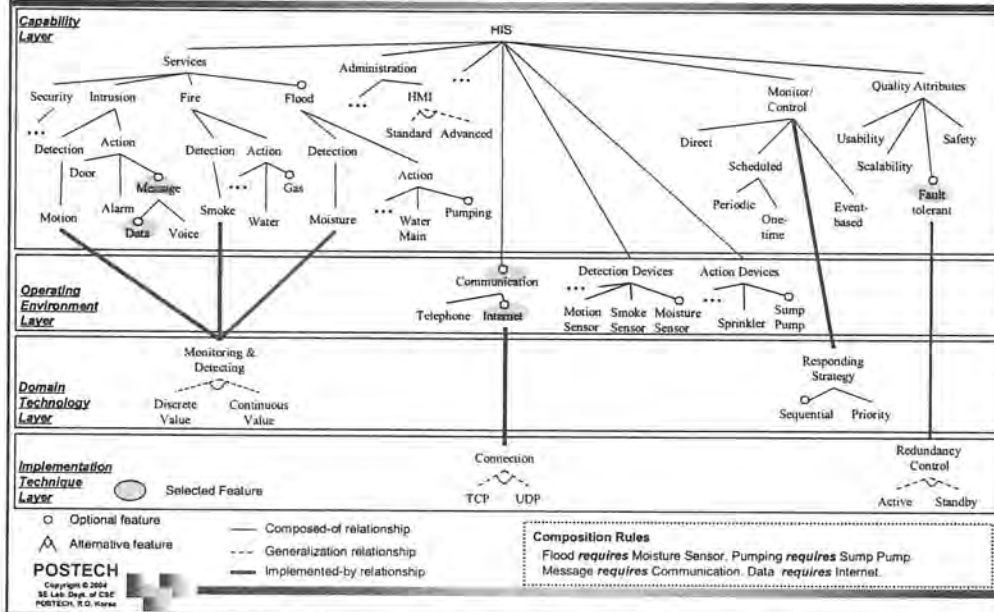
What is Product Development?

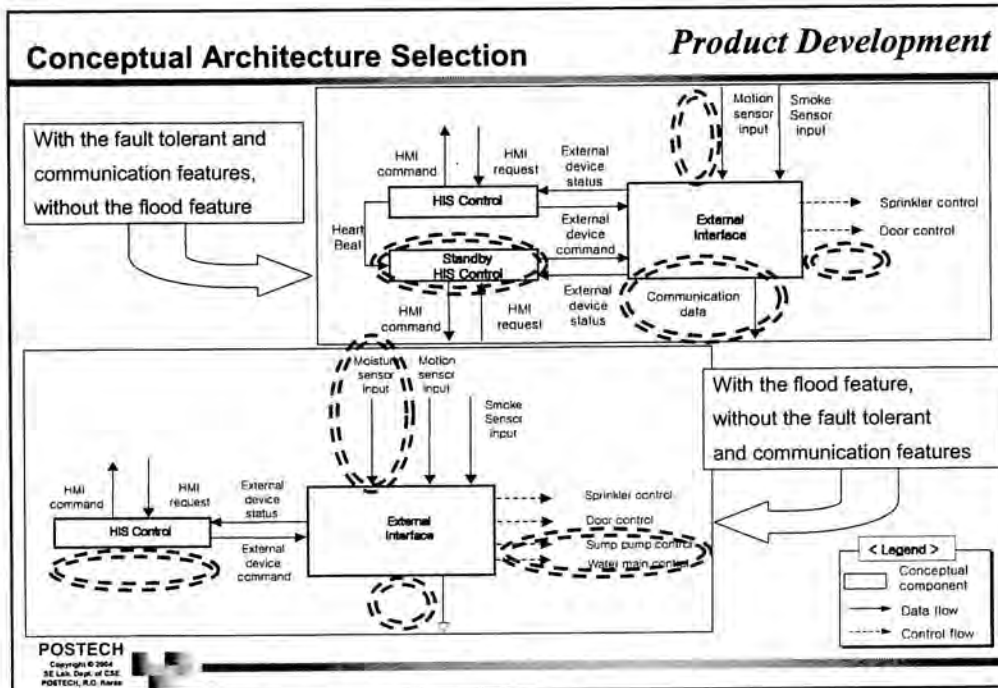
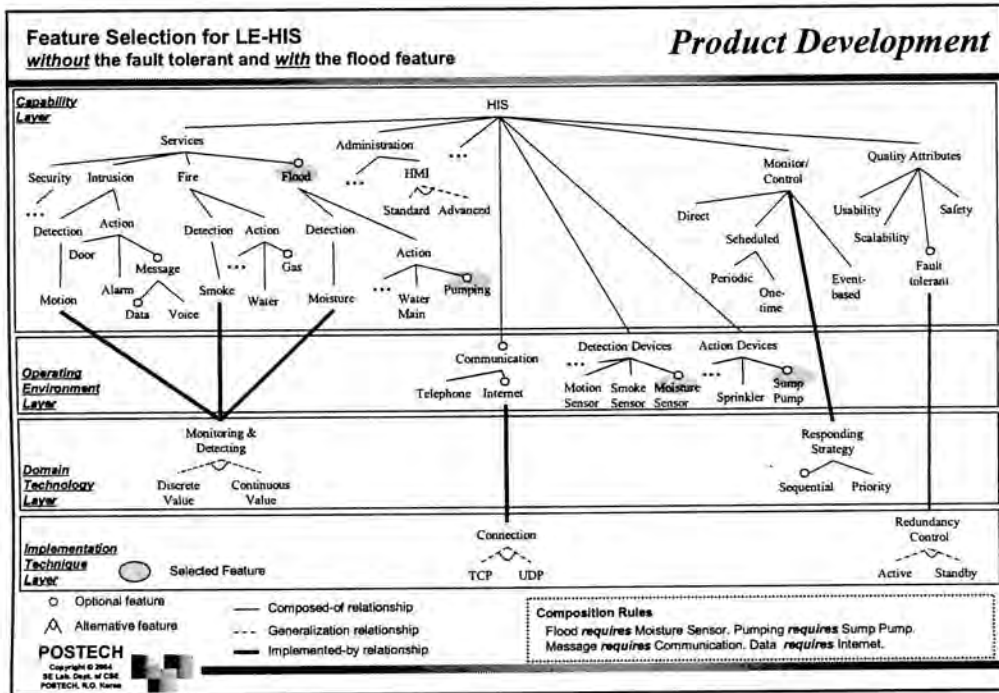
Product Development

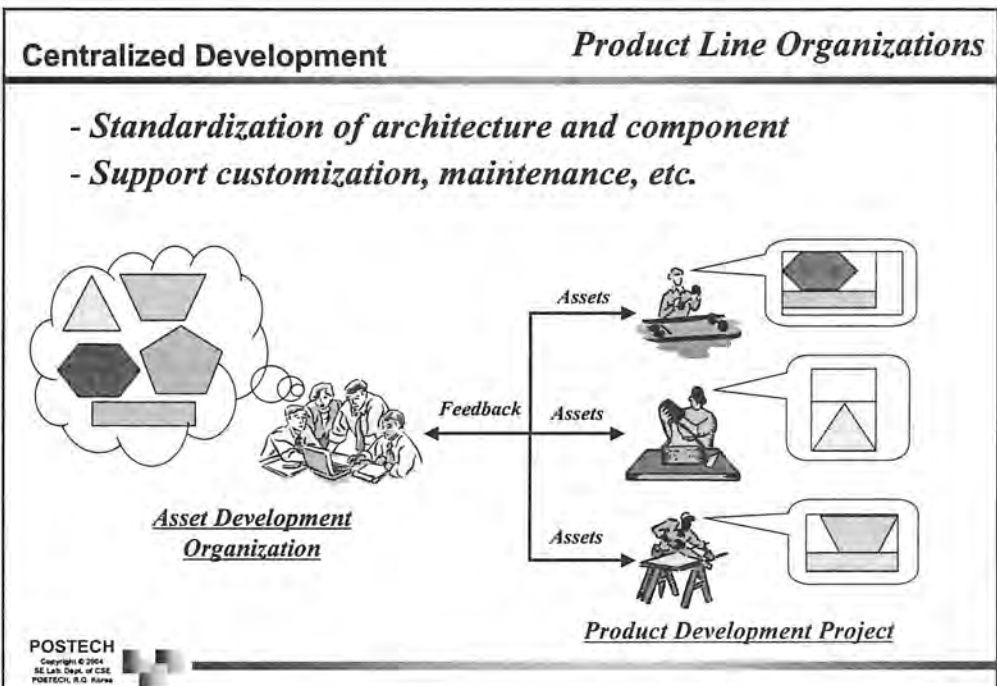
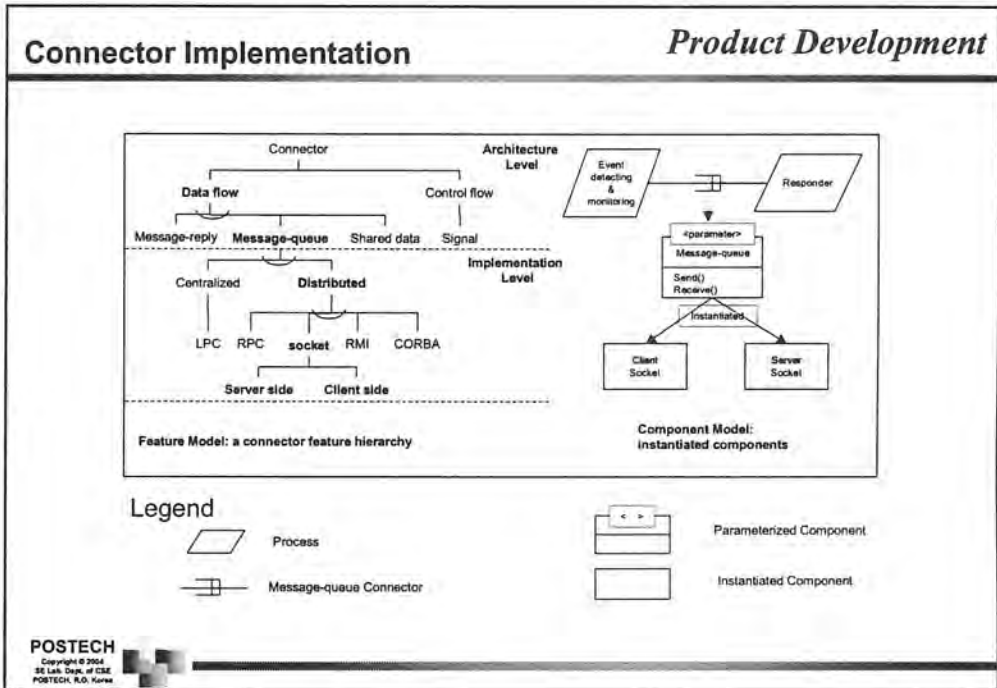
- Product development is a process of developing a specific product making use of the product line asset developed during the product line asset development process.
- Product development proceeds by:
 - analyzing user's requirements,
 - selecting appropriate and valid product line features from the feature model,
 - identifying the corresponding architecture models,
 - completing the product development by reusing software components, and
 - adapting components as needed.

Feature Selection for LE-HIS with the fault tolerant and without the flood feature

Product Development









Centralized Development

Product Line Organizations

- Advantage

- Product line-wide sharing of engineering vision
- Efficient use of corporate expertise
- Systematic management of assets

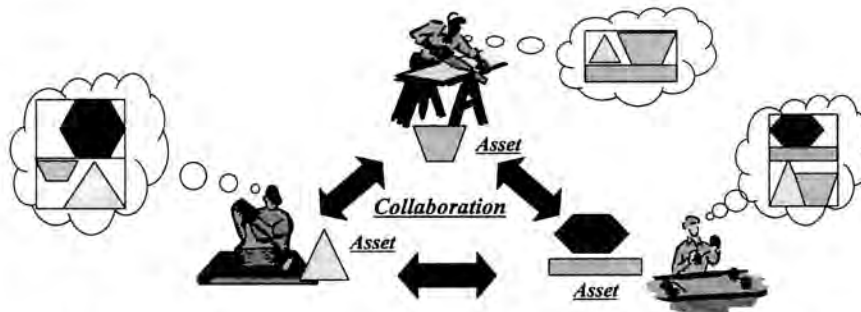
- Disadvantage

- Large up-front investment and difficulty to justify the initial cost
- No immediate return for investment

Collaborative Development

Product Line Organizations

- *Asset development and support responsibilities are distributed*
- *Each project develops asset components and shares them with others*





Collaborative Development

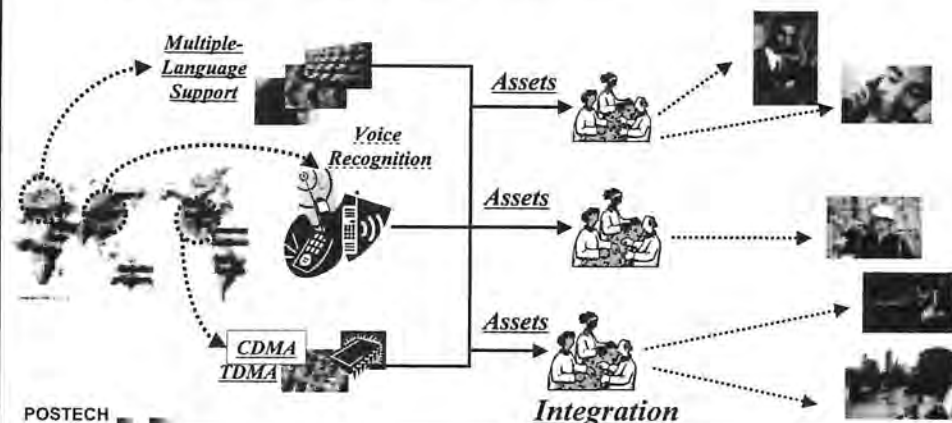
Product Line Organizations

- Advantages
 - Less overhead cost
 - Asset development costs are distributed
- Disadvantages
 - Coordination difficulty
 - Need a convincing cost / benefit model
 - Hard to identify others' needs

Distributed Development

Product Line Organizations

- *Asset development and support responsibilities are distributed.*
- *Each organization has independent budgeting and accounting*
- *Each project integrates asset components*





Distributed Development

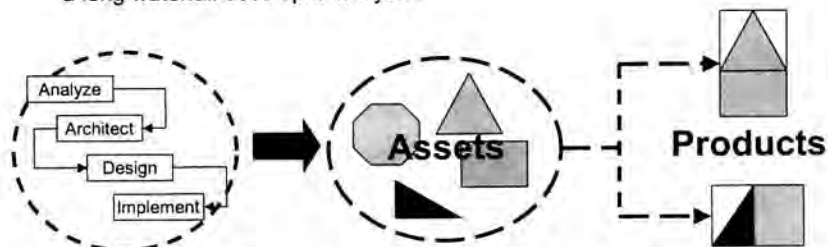
Product Line Organizations

- Advantages
 - Expertise in specific areas
 - Multi-national, geographically dispersed organization
- Disadvantages
 - Hard to identify others' needs
 - Collaboration difficulty

Proactive Model

Product Line Process Models

- Proactive model (waterfall model)*
 - The proactive approach to software product lines is like the waterfall approach to conventional software. You analyze, architect, design, and implement all product variations on the foreseeable horizon up front.
 - This approach might suit organizations that can predict their product line requirements well into the future and that have the time and resources for a long waterfall development cycle.



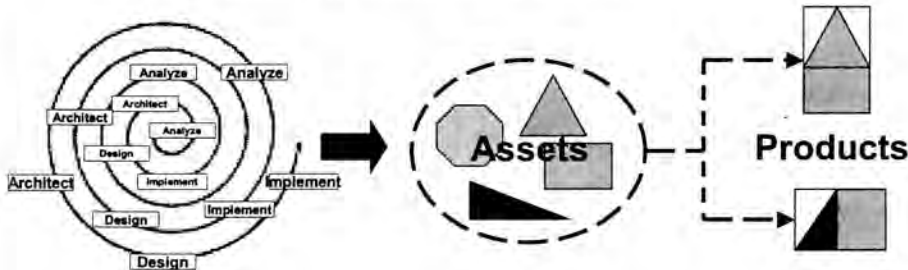
* Charles Krueger, "Eliminating the Adoption Barrier", IEEE Software, Jul/Aug, 2002, pp. 29-31



Reactive Model

Product Line Process Models

- Reactive model (spiral model) *
 - The reactive approach is like the spiral or extreme programming approach to conventional. You analyze, architect, design, and implement one or several product variations on each development spiral.
 - This approach works in situations where you cannot predict the requirements for product variations well in advance.



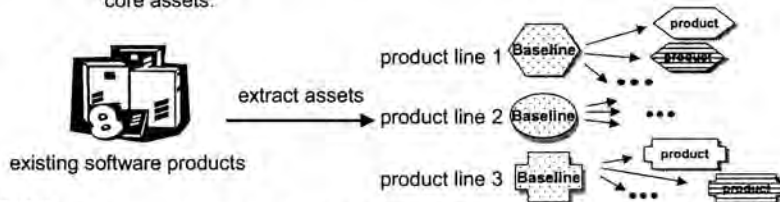
POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, K.O. Korea

*Charles Krueger, "Eliminating the Adoption Barrier", IEEE Software, Jul/Aug, 2002, pp. 29-31

Extractive Model

Product Line Process Models

- Extractive model*
 - The extractive approach reuses one or more existing software products for the product line's initial baseline.
 - Require lightweight software product line technology and techniques that can reuse existing software without much reengineering.
 - Effective for an organization that wants to quickly transition from conventional to software product line engineering
 - This approach *does not* support the possibility of one organization developing the core assets and a separate organization developing the products based on the core assets.



POSTECH
Copyright © 2004
SE Lab, Dept. of CSE
POSTECH, K.O. Korea

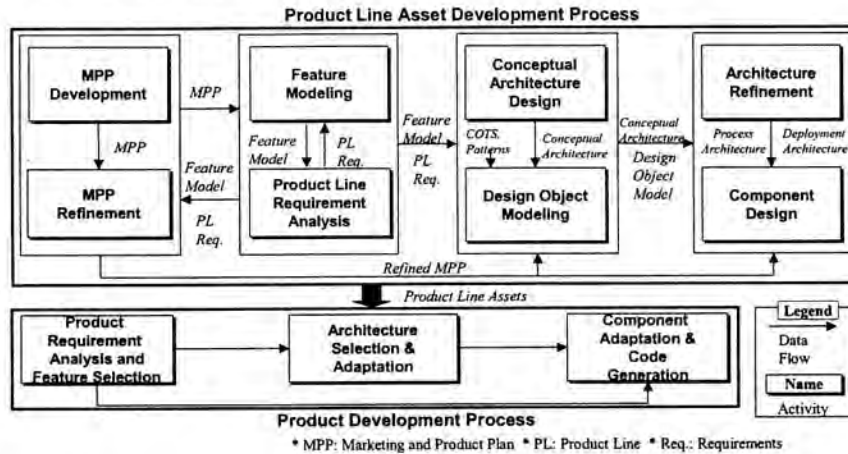
*Charles Krueger, "Eliminating the Adoption Barrier", IEEE Software, Jul/Aug, 2002, pp. 29-31



FORM

Examples of Proactive Models

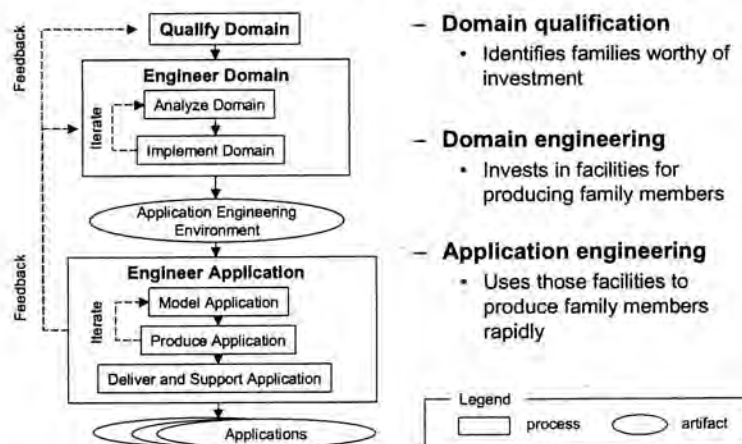
• FORM (Feature-Oriented Reuse Method)

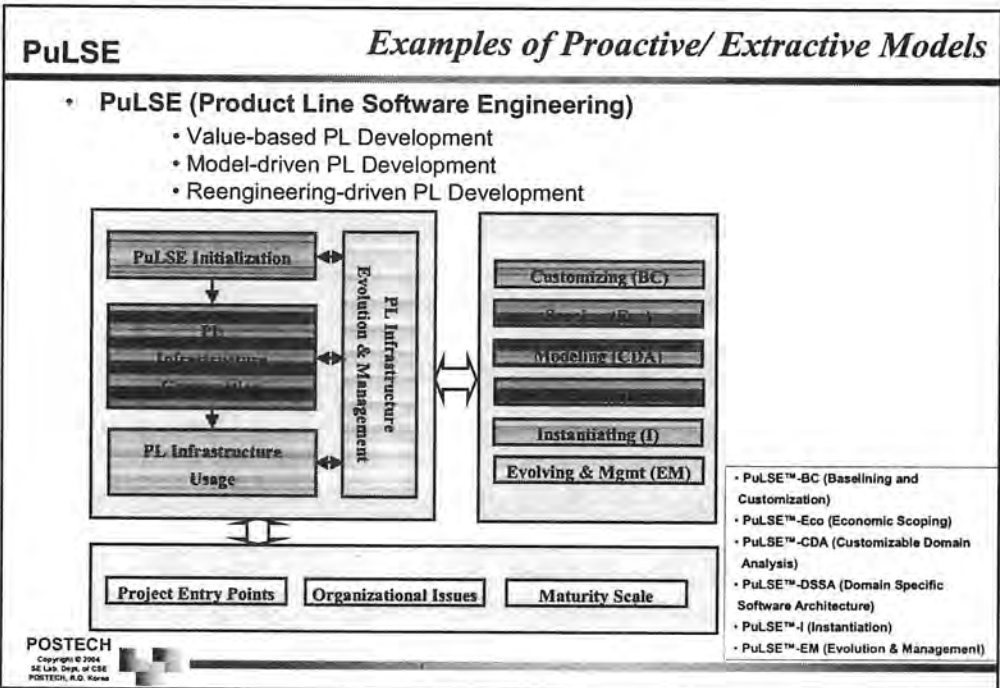
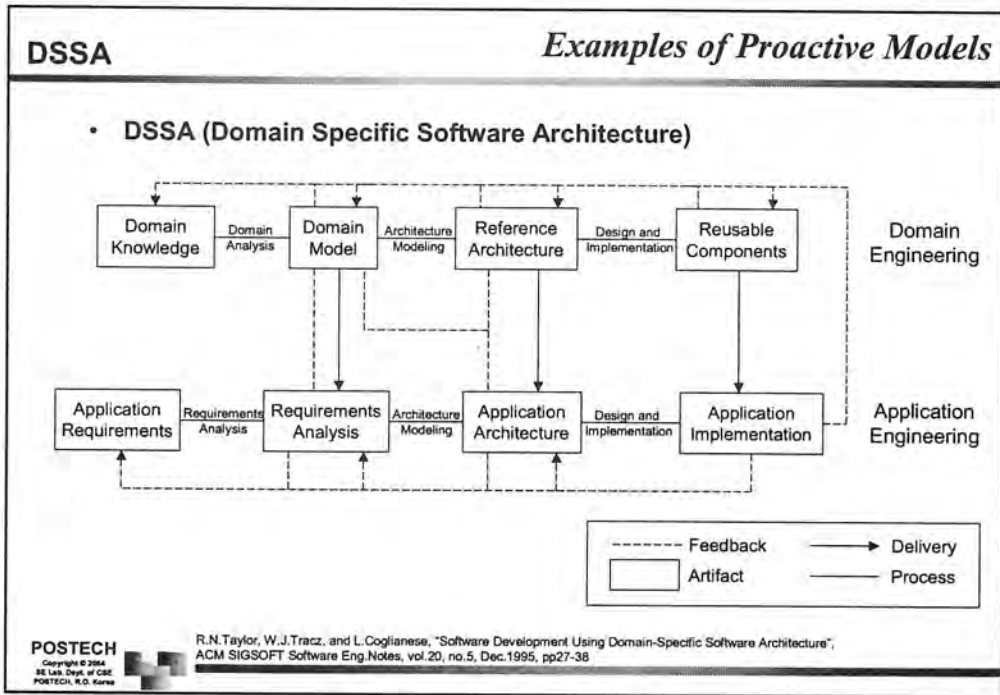


FAST

Examples of Proactive Models

• FAST (Family-Oriented Abstract, Specification, and Translation)



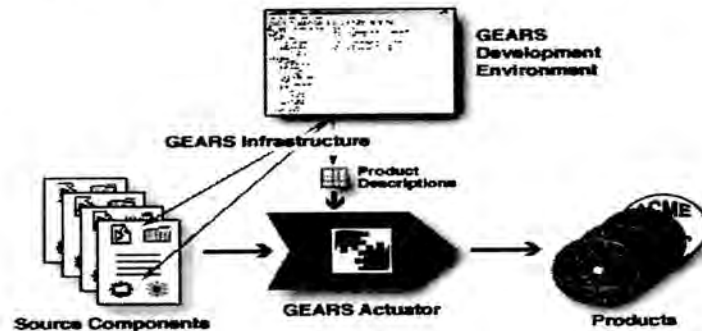




GEAR

Examples of Extractive Models

- **Commercial Product**
 - "GEAR" – BigLever Software inc.
 - A software mass customization engineering environment



Other Issues

1. *Return on investment*
2. *Availability of domain expertise*
3. *Evolution of product line*
 - *Rapid technology evolution*
 - *Dynamic market environment*
4. *Education of the technology*
5. *Availability of tools*

SOBA フォーラムの報告

SEAMAIL 編集部

はじめに

過日(2004.02.13)、東京で開催されたSEAフォーラムは、参加者33名という盛況で、無事終了した。最近、SEAMAILに、報告記事が出ないのは、ちょっと寂しいと思い、主に当日の全体討論の様子をご紹介します。

1 開催の概要

昨年(2003年)、「銀座スエヒロ本店」があったところに建てられた「全国情報サービス産業厚生年金基金会館」の7F・会議室Bで開催された。この会場が選ばれたのは、「インターネットに繋げたいのデモを見てもらいたい」という講師からの要望で、そのような設備がある場所、という理由である。とてもきれいで立派な設備のビルである。SEAのイベントとしては、昨年の暮れのSEA-SPIN分科会での利用以来、2回目である。

最初に、関西からお招きした3人の講師の方に、講演およびデモをお願いした。

1. 13:30~14:30

「SOBA フレームワーク：ブロードバンド時代のソフトウェア基盤の開発」

講師：角田 誠 さん(NTTコムウェア株式会社)

2. 14:50~15:50

「Eclipse上で遠隔でのペアプログラミングを可能とした開発環境 Sobalipse」

講師：林 良生 さん(オムロン株式会社)、新納 正康 さん(財団法人京都高度技術研究所)

これを受けて、最後の1時間(16:00~17:00)は、参加者全員で、意見交換を行った。ここでは、その意見交換の様子を報告する。

2 全体討論から

小林(SRA)：今日の講演を聴かれて、会場の皆さん、それぞれに、SOBAについて、これからの期待、要望、意見などをお持ちだと思います。それらを是非、忌憚無く出して頂いて、SOBA フレームワークをこれから発展させるための材料になればと思います。

中野(大阪市立大)：ライセンスの関係について、もう少し教えて下さい。特に、オープンソースでは、色々なライセンスの考え方があると思うので。

角田(NTTコムウェア)：現在、LGPL¹のものしか(SOBAでは)ありません。LGPLとは、ソースに直接、手を加えた人は、それを公開して下さい、というものです。但し、SOBAを利用して、SOBA

¹<http://www.gnu.org/copyleft/lesser.html>

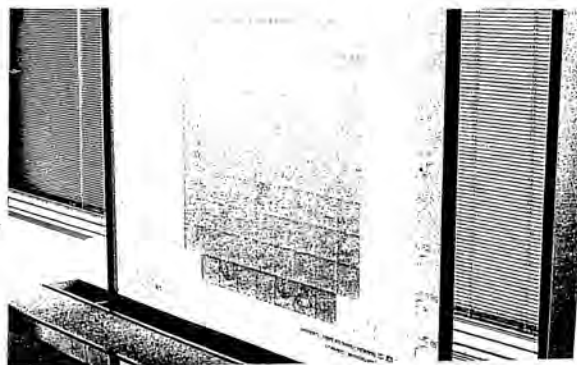
の上にアプリケーションを作った方は、御自分で作られた部分に関しては、ソースを公開する必要はありません、というのが LGPL といわれるライセンスです。

中野：アプリケーションの権利は誰のものになるのですか？

角田：それは、アプリケーションの作者になります。

鈴木(コンサルタント)：今日のご講演は、開発環境という、真面目な、ちょっと硬い話だったと思うのですが、ネットゲームのプラットフォームとして、といった使い方は考えられていないのでしょうか？

角田：実は、SOBA の配布セットの中に、1つ、ゲームがあります。簡単な対戦型のゲーム(写真参照)です。これは、昨年、SOBA プロジェクトに参加してくれた、フランス人の学生が作ったアプリケーションで、迷路を探索しながら、対戦が出来るというものです。ちょっと凝っているのですが、若干、レスポンスが遅いのですけれども…。



鈴木：SOBA を、携帯電話 FOMA とリンクさせて使う例を見せてもらいましたが、グラフィックなどもきれいなので、SOBA をプラットフォームにして、FOMA の上のゲームソフトを作るとヒットするように思うのですが…。

角田：我々も、携帯電話をどういふふうにも活用していこうかと、考えています。是非、企画、共同開発の話などありましたら、営業の方へご連絡ください。

A：その場合、スケーラビリティは、問題にならないのでしょうか？

鈴木：いや、それが P2P の長所だと思うのです。サーバが不要なのですから。

角田：但し、携帯電話の場合、問題になるのは、FOMA の場合ですと、通信出来る相手というのは、ある程度固定されてしまいます。ですから、携帯電話の場合は、完全に P2P の通信が出来ませんが、1つのサーバを介して、という構成になると思います。しかし、それでも、ゲームシステムの全部を1つのサーバに収容して処理するというわけではありませんから、ある程度サーバを分散させて、そのサーバが P2P で通信を行うので、スケーラビリティという点では、高いものが出るのでは、と考えています。

鈴木：そのとおりで、既存の携帯電話のアプリケーションの実現方法ですと、一度に何十万というアクセスがサーバに来ることになります。端末数が、数千万あるわけですから。それを P2P で、それが実現出来るとなると、その心配は無くなるはずですが、但し、(従来の携帯電話の課金のメカニズムを使えないと思うので) 新たな課金システムをどうするのか? といったビジネスモデルは、これからの検討課題だと思います。

A：対戦型のゲームだったら P2P というモデルは有利だけれども、「多人数でじゃんけんをやる」というようなゲームは難しいのですか?

角田：いえ、HTTP サーバとしての構成は、従来通りになりますが、その裏に SOBA のフレームワークを乗せたマシンを配置し、分散させることで、その部分を P2P で分散させることが可能だと思います。

河野(琉球大)：現在の SOBA のスケーラビリティはどれくらいなのでしょう?

角田：スケーラビリティをどのように計るか?、については、難しいと思うのですが

河野：Sobalipse で何人くらい? までとかの経験値はありませんか?

角田：テレビ電話のシステムで、すべての人がビデオカメラを共有したアプリでは、9 人までは、使用した経験があります。それも、実験の際に、使用出来たカメラ機材の数が 9 台だったというだけで、それ以上でも、実用上、問題ないと思われれます。また、これが、メッセージングのみ、といったアプリであれば、もう少し多い人数でも、いけるように思います。

河野：SOBA は、プロトコルレベルで仕様をフィックスさせているのでしょうか? それとも、Java の API レベルでしょうか?

角田：現時点では、Java の API レベルで、プロトコル部分については、フィックス出来ていません。また、現在の通信プロトコル部分は、Java の実装に依存していますので、それらも含めて、改めて API も含めて、再設計しようとも考えています。特に、通信プロトコル部分の標準化は、重要と考えていますので、今後、これに取り組んでいきたいと思っています。

河野：通信プロトコルが、Java に依存しているというのは、どのあたりのことですか?

角田：オブジェクトを、直接、シリアライズして送っている部分が、Java に依存しています。ですので、当然、クラスを変えると通信できなくなるという問題があるので、美しくないと思っています。ですので、オブジェクトのシリアライズのようなことをせず、他の言語(例えば C でも) ちゃんと動くような通信プロトコルを規定したいと思っています。

小林：Sobalipse で、プロジェクトの最初から最後まで、ずっと共有空間を立ち上げておいて、というほど大げさなことでは無いのですが、自分の仕事でも Sobalipse を使おうとしているのですが、共有をするときに面倒だと思うのは、自分がまだ、十分に使い方を熟知していないこともあるのだと思うのですが、自分の方で、グローバルモードにして、セッション上、共有できるようにしておいて他の人が、それを開いて、それを共有するというときにですね、セッション上で共有したものを、一覧の中から選べるような状態になっているのでしょうか? ちょっと、そこらあたりの使い方で、私の所ではもたついておまして、今、どうしているかといいますと、共有する方でも同じファイルを開いてそれをグローバルモードに切り替えて、という使い方をしてるのです

林(オムロン)：最初は、開発の立場でも、そのような使い方の良いのかなあ、と思っていました。ただ、極力、モードという概念は、入れたくなかったのです。実際に、ご覧にいれましょう。まず、Sobalipse を立ち上げてセッションに入るところまではやっていただく必要があります。そして今、「自分が共有しているファイルを相手にも共有してもらいたい」という時に「メッセージ機能」を使って、メッセージを相手に送信することが出来ます。ここで「エディタへのリンクを挿入」とすると、そのリンク情報が相手に送信されますので、受取った人は、それをそのまま(そのリンクを)クリックしてやることで目的のことが出来るかと思っています。このようなリンクの機能を、チャット

の中に埋め込んだりすることが出来ます。このような機能を追加したので(他の人と同じファイルを共有するという)最初のとっかかりがスムーズに出来るようになったかなあ、と思っています。

また、この機能とは別になりますが、一度、共有を開始してから、相手の操作を追いかける目的で「追従機能」というのがあります。これを使うと、今、相手がどこら辺を操作編集しているのが容易に判ります。

このように、今は、出来るだけ簡単な操作で、かつ、出来るだけ汎用的で重要な機能から実装するように作業を進めています。ですので、もし「こういう機能があれば」というご意見がありましたらお聞かせ下さい。

小林：ありがとうございます。また、話題は、変わりますが、先ほどのようなインターネット上の環境で、セッションを共有するといった場合に、ファイヤーウォールを越えるとか、中継サーバを置くとか、LAN上とは異なった色々な制約とか条件とか出てくると思うのですが、その場合に、どれくらいパフォーマンスとかスケーラビリティとか、また実証実験的なことがされているか情報がありましたらお願いします。

角田：中継サーバを使った通信というのは、スケーラビリティという観点からは、まだあまり評価していません。ただ、経験的には、1台の中継サーバを介して画像データなど(大量のデータ)を送るのは、現時点では(クライアントが)数台が限度です。どうしても、今の方式ですと、中継サーバが、P2Pで線を張る形になってしまいますので、どうしてもネットワーク帯域が厳しいものになると思います。但し、インターネット上で、どれくらい使えるのか?ということですと、一昨年に中国とアメリカと京都を結んで実験をしたという例はあります。また、現時点では、中継サーバは、まだあまり積極的に評価、チューニングなどを行っていないとも云えますので、これからどんどん性能的にも上がっていくとは思いますが、今後にご期待下さい。

小林：私は、LAN上での環境での開発しか考えていないのですが、実際に、このSOBAフレームワークによるサービスをインターネット上で展開するとすると、現状、何か課題になるような点、あるいは、乗り越えなければいけないようなものは無いのでしょうか? また、実際に運用側の設備、あるいは、体制を整えていけば、ビジネスとして発展していける環境が、現状、整っているのでしょうか? ISP(インターネットサービスプロバイダ)の立場からだとどのように見ておられるのか、ご発言頂ければと思います。深瀬さん、いかがでしょうか?

深瀬(フリー)：昔、ちょっとインターネットのサービスの仕事をしていまして…(笑)。多分、ご質問の主旨は、今のインターネットのトラフィックが、どれくらいのボリュームなのか?。また、プロバイダ側で、どれくらいのスケーラビリティが想定されているのか、といったことにご興味がありだろうと思います。正直言いまして、通信回線自体は、すかすかなんですよ。実利用率は、多分数パーセント、それも下の方の数パーセントです。逆に、データセンター側のサーバの処理能力問題であるとか、ルータの問題であるとか、そのような機器が、回線のキャパシティに見合ったものになっていないのが実情です。で、同然、新しいメカニズムの機材は開発・発売されますが、それよりも先に伝送路の方が速度が上がっているのが実情です。ムーアの法則(18ヶ月で、処理速度が2倍になる)がありますが、通信回線についていえば、ここ数年、2年で100倍になっています。ユーザへの直近の(ラスト1マイル)もここ数年、高速になり且つ低価格になっているのは、皆さん、ご存知のとおりです。ですので、その周辺機器が、回線の高速化に十分に追従出来ていないことが、このような新しい共有関係のミドルウェアが登場してきて、それを使ったアプリケーションを作ると、ミスマッチの部分が、ヘビーロードになってお客さんから苦情が来る、ということは直ぐに想定出来ると思います。

で、昔、我々がインターネットで、宣伝材料として考えたときに、ネットワークゲームというのも当然、考えました。例えば10万人規模のユーザが居るといったものの実現可能性をです。その場合

に、やはりサーバ側やルータの問題がネックになっていました。その状況は今もさほど変わらないはずですが、ですので、そこら辺が解決しないと、難しいと思います。

また、ビジネスとして考えた際に、課金をどうするのか?も問題でしょう。携帯電話の場合には、携帯電話のサービスを提供する会社が課金代行というサービスを引き受けてくれるのですけれども、彼らが考えている課金モデルとこういう共有型のサービスモデルの課金モデルは、多分、まったく異なるものだと思いますので、その点についても知恵を出さないといけないと思います。

以上の2つの点から、アンバランスな点を解決する目途が半年以内というレンジでは難しいように思います。



小林：どうもありがとうございました。他に、いかがでしょうか?

河野：「オープンソースの共同開発ツール」と紹介されていましたが、そう (Sobalipse) を捕らえると、直接的に競合するのは、CVS²なのではないかと思っています。CVSとは、どのように関連付けるつもりでしょうか?あるいは、CVSよりも、この部分は良いとかありましたら教えて下さい。

林：CVSとの違いは、リアルタイム性を重視している点だと思います。相手が変更している箇所が、そのまま見えるという点が、大きな違いだと思います。使い方としては、作業を始める時に、CVSからデータを取ってきて、それを Sobalipse の上で、共同作業をして、また最終的な成果物は CVSに入れる。これで、十分、棲み分けが出来るだろうと思っています。いかがでしょうか?

河野：私は、棲み分けては、いけないのではないかと考えています。協調、つまり、相互に補うような形で、ものづくりをしないといけないのでは?このままでは、絶対に CVSには勝てない、と思います。

小林：私も普段、実務では、リポジトリ管理・開発の管理は CVS を使っています。で、時々、これはちょっと目をかけてやらないといけないなあ、あるいは、ここは相談しながらやってください、という場合には、Sobalipse を使って作業してもらうように指導しています。その際に、競合というわけではないですが、CVS のリポジトリの為にすっきりしない面があります。Sobalipse だと、皆がそれぞれ勝手に、ファイル f1 は Xさんと共有していて、ファイル f2 は Yさんと共有していて、という作業になります。すると、何所にどんなバージョンがあるのが、段々、把握し辛くなってきます。で、それを解決するために、一旦、きれいにしなければいけないから、CVS にアップデートして

²Concurrent Version System (<http://www.cvshome.org/>)

コミットしてもらおうといったことを、作業している人をお願いすることになります。そして、そのときに(整合性に)矛盾が出てきたらどうしよう、と思ったりします。ですので、Eclipseが(例えば裏で)こまめにCVSに対して更新をかけるとかそういった機能もあっても良いのではと思いますし、また先ほどご指摘のあった用に、本当にCVSの提供する機能までカバーしてしまうのも面白いのかもしれませんが。

藤枝(北陸先端科学技術大学院大学): CVSとSobalipseは、全然、異なるものなので、連携する必要も戦う必要も無いと思います。そもそもCVSは、構成管理システムなので、合意が形成された変更・履歴を格納してゆくものです。一方、ここで使われているSobalipseは、変更を作り出してゆく作業を支援するツールなので、まったく違うものだと思います。ですので、連携はしても良いとは思いますが、戦う必要など何も無いと思います。

林:(Sobalipseのベースになっている)Eclipseには、CVSのプラグインが既に存在していますので、Sobalipseの裏でCVSを動かすのは、さほど難しくないように考えています。

藤枝:あと、構成管理ツールの話になったので…。私の知り合いのイタリアの研究者が、P2Pの構成管理(バージョン管理)システムを作りました。これは、pirewareという名前で、JavaのデータをP2Pで共有するというフレームワークを使って作ってあります。これの特徴は、リポジトリのサーバを一々用意するのは面倒なので、リポジトリの中身は、全員保持していても構わない、という発想です。ですので、CVSのバージョン木のようなものを、Sobaでいうところの「共有」している形になっています。で、彼(開発者)は、学生のプログラミングの演習に使用しているようです。これならば、チーム毎に各々CVSサーバを用意するといったことが要りませんのでチーム毎にP2Pのリポジトリを使って作業をすることが容易に出来ます。さらに、家に帰って各自が変更修正したものを、また学校に来て、同期させています。そして、その同期を破綻無く行う仕掛けについては、論文があります。

林:参考になります。ありがとうございます。

小林:他に、いかがでしょうか?

河野:課金というのは、将来的に考えておられるのでしょうか?

角田:課金(の仕掛け)が必要だろう、という意見は、以前からあるのですが、課金自体は、アプリケーションに任せよう、と考えています。ただ、(Soba上のアプリケーションから)課金のための情報を(Sobaの中から)取り出す仕組みだけは、用意しておく必要があると思っています。但し、今現在のSOBAの開発の中での重要度は、あまり高くありません。というのは、そのようなものを必要としている案件が少ないからです。

河野:単純に、シェアウェアにするとか、そういう考えは無いのですか?

角田:今は、そのようなことは特に考えていません。

B:今の質問内容は、課金といっても、SOBA自体が有償化するか?、とか、シェアウェア化といった話題ですよね?。で、その前の角田さんのお答えは、課金のメカニズムとして、アプリケーション側で、どのような仕組みを作るか?といった話でした。が、ベースとなるSOBAは、現在、LGPLで提供されているということですが、将来的に(部分的にでも)有償化するとか、あるいはシェアウェア化するとか動きはどのようなのでしょうか?と言いますのも、多くの皆さんは、会社組織に属されているので、何らかの形で利益を追求しなければならないと思います。ですので、差し支えない範囲で、今後の展開についてお教え頂ければと思います。

角田:ビジネスモデルとして、どういうところで利益を得るかというのは、非常に重要だと思います。で、SOBAフレームワークの基本的なところは現状通り、無償で、と思っています。それ以外のアドオンの部分については、もしかしたら有償で提供というものもあるかもしれませんが。例えば(これは例で、そういうふうにしようと思っている訳ではないですが)Windowsの画面の共有機

能が、新しく出来たとします。そうすると、その部分はアドオンという形で有償で提供させていただく、といった選択も出来るかなあと思っています。また、SOBA 自体にアプリケーションを乗せることが出来るわけですが、その作業のサポートなどを (例えば SOBA プロジェクトとして) 有償で行うというような形も考えられると思います。

C: 既に、現在の SOBA が、LGPL で配布されているわけですので、ある時点で、それが外れて有償になったとすると、フリーソフトウェアを開発している人は、現在の LGPL のバージョンをベースに新たな (機能追加、バグ修正をされた) フリー SOBA が (正規版とは別に) 出来てしまう気がします。ですので、今現在、開発されている方々に敬意を表しつつ、今のままの LGPL を維持するのが、お互いが一番幸せなように思います。

角田: そうですね。やはり、フレームワークを普及させるということが、一番、大きなところがあるので、フレームワークの基本的な部分はこれからもフリーで提供してゆくのが正しいのかなあ? と思っています。その代わり、全てフリーのバージョンしか無いというのではなくて、例えば、企業様向けには、有償のパッケージを用意して、そこにはサポートも付けますよ、といったような売り方もあると思うのですよね。例えば、全て LGPL で出してしまうと、(特定) 企業向けにカスタマイズが必要になった場合、それも公開しないといけない、というようなことになると企業さんがなかなか (SOBA に) 手を出し辛くなってしまいます。そういった場合には、デュアル・ライセンスのような形で、企業様向けには有償で販売していく、ということも考えられるかと思っています。

河野: 私は、LGPL は、正しい選択のように思います。かなり緩いですし…。また GPL と課金は、理論的に両立しますので、今の形態 (LGPL) のままで良いのではないのでしょうか?。例えば、Linux の場合には、サービスでお金を得るということも出来て、それは GPL に矛盾していませんから。

角田: 只、人的な資源も限られたものがありますので、どのようなところでビジネスにしてゆくの、一番良いのか、というのは、これから真剣に考えていかなければならないところだと思っています。是非、良い知恵がありましたら、プロジェクトにお教えください。

D: 今日、いくつか SOBA の機能を見たのですが、e-ラーニング系にそのまま使えるような機能が、すでに相当多く組み込まれているように思ったのですが、この場に教育関係者の方も多く居られますが、社内教育とかトレーニングなどで、これを実際に使われている例は無いのでしょうか?。あればご紹介ください。

角田: 教育分野では、一番古くからやられているのでは、二村先生 (早稲田大学) の研究室で、遠隔教育のシステムを SOBA フレームワークで作っています。これは、ブラウザをアプリケーションとして共有し、その中にフラッシュで作ったコンテンツを、それぞれの PC 上で表示させて、講師の人がスタートさせたりストップさせたりといったことを出来て、その情報を共有することが出来るというものです。また、京都造形大でも、別の SOBA を使ったものを作られているようです。

小林: 先ほど、海外との接続実験のことについて話がありましたが、何か、海外との連携で進んでいるようなものはあるのでしょうか?

角田: 先ほどのアメリカ、中国と京都を結んでの接続実験以外に、先ほどの早稲田大学のプロジェクトでは、コンテンツはアメリカの会社が作り、それを使って共同でアプリケーション開発をしていると聞いています。しかし、SOBA プロジェクトとしては、まだ、海外との提携というのはありません。

E: Sobalipse でのペアプログラミングでの経験談があれば、ご紹介ください。

林: 実は、良い意味で協調作業が出来れば素晴らしいだろうなあと考えて作り、使っているのですけれども、どうも周りに支持者が少なく、あまり体験談がありません。自分としては良く出来ていると思うのですが…。仕事中にやっているのではないかと、その非難が周りにありまし

て…。の評判はかんばしくありません。

E:プログラマの性格とか会社の風土にもよると思いますが、自分のやっている仕事の過程をオープンに出来るか否かも、この Sobalipse が支持されるか否かに随分と影響すると思います。

河野:昔(1980年代)は、当時のパソコン通信のBBSで情報を共有することで、プログラムを開発・改良してゆくという事例がありました。そのようなことからそのようなツールとして(当時ならば)BBSで十分であったと思います。しかし、その時にも Sobalipse があったなら、便利だと考え使うでしょう。ですので、ツールがあって、それでコラボレーションをするのではなく、先にコラボレーションがあって、それをサポートするツールがあるのだと思います。この Sobalipse が、使えるかどうかは、これを持っていて、仕事中には使用してはいけない、というような会社では、絶対に役に立たないでしょう。

鈴木:このツールは、先生と生徒、というような、教える人と教わる人という関係がハッキリしているときに、すごく良いものだと思います。もし、同程度の人が使うと、絶対に喧嘩になると思います。僕は、ペア・プログラミングには、懐疑的です。例えば、習字を考えましょう。下手な人が書いたものに、上手な先生が赤字で直しを入れるとか、手を持って直してくれる、というのは非常に良いことだと思うのですが、同じくらいにうまい人が、「俺はこう書くのが良いと思う」と言われても「いや、ちがう」となって、収拾がつかないと思います。自分がプログラミングして、わからないときに、R. ストールマンに電話をして尋ねて、即座に「こうプログラミングするんだ」と言われれば、それを受け入れますが、同程度の人から同じことを言われたら、絶対に受け入れないと思います。ですので、他人に教えるときに、こういうツールがあると良いと思います。また昔、実際に、判らないところをすぐ近くにいる先輩に「すいません、ここ、どうやって書くんですか?」と聞いた経験からみても良い。で、昔なら、そういう先輩が近くにいたわけですが、今では、それがインターネットとこういうツールを使って、早く、良いコーディングをする技術を教えられるのではないかと思います。

小林:ペアプログラミングということですと、私も他人にとやかく言われると、ムカッとくる性格なので、同じようなことを感じるんですけども、スキルとかは別にして、あるプロジェクトに、新しいメンバーが入ってきた場面を想像して下さい。先にプロジェクトで、例えば、フレームワークやアーキテクチャなどを、選んだり整備して、「このアプリケーションは、大体、こういう機能でこういう作り方でやろう」という土台作り・地慣らしがされてあったとします。そうしたときに、新たに入ってきた人に、このフレームワークで、プログラミングの(作業の)仕方を教えるのに、ただ単にサンプルを渡して教えるのと、実際に、自分で「このフレームワークを使う場合にはどういう順番で…」というふうなことを考えながらコーディングして、どこでコケルか?といったことを、実際に操作して見せられるのとでは、効果が違うなあと思います。ですので、「Sobalipse が無いと作業が出来ない」というわけではないのですが、あると効果があるなあと思っています。ですので、ペア・プログラミングを「同じ1つのものを2人で検討しながら作る」ということとは別に、ある人が持っているイメージとかモデルというものを、別の人に伝える、あるいは、別の人がある人の仕事の仕方(の過程、物が作られる過程)を見ることは、ただ単に出来上がったものを見るよりも、はるかに効果的に伝わると思います。大人数ではなく少人数、例えば3~5人程度で、1つのテーブルで1つの画面(ディスプレイ)を囲んで、チームの中のアーキテク的な人がコーディングしているのを見て、その中で「ちょっとここのは、自分でやってみて下さい」といったことをすると、かなり、効果的に、開発のイメージを共有するのが効率が良くなると思っています。ですので、私は、そういうところで、期待しているところがあります。

F:私は、プログラミングよりもドキュメントを書くことが多いのです。で、思ったのは、Sobalipse で共同で書けるもの(対象)は、おそらくプログラムだけではなくて、普通の業務で他人と共同で文

章を書く、といった協調作業を行うのに効果的に使えるような気がします。もし、Sobalipse をもっと広く普及させようと思うのなら、テキストエディタとして2人で、協調して作業を進められますよ、というのは、良い例にはなるのではないのでしょうか?。まず、そのような仕事の母集団が、多いと思います。

と言いますのも、私が大学生の時に、服飾関係のメーカーでアルバイトをしていたのですが、その時に、広告を企画することを4人ほどのチームで行いました。最初はメールと電話で「1ページ目は私がやるので、2ページ目は、xxさん、お願いします」「2ページ目のこのところは、バランスがどうも良くない」といったやり取りをしていたのですが、こういったことを Sobalipse を使ってやれたら、ずっと効率的に作業が出来たと思うのです。

藤枝：さきほど CVS との棲み分け、といった話があったかと思いますが、えっと、何を言いたいのかといいますと、「anonymous(匿名) でコミットも出来る CVS」があったとか、「Wiki のようになって誰でも書けますよ、という、そのようなバージョン管理システム」があったとしたら、それらとの差異、特に「複数の人が同時に変更が見られる。これが良い」という Sobalipse の特長がより具体的に無いと、売りにならないと思います。

そのような意味で、そのような変更(差分)を見せる仕掛けについては、何かお考えになられていることはありますか? と言いますのも、「こういう意図で、このように修正しました」という“修正の結果”が知りたいという人もいますし、一方で、その“意図”が知りたいという人もいます。また、単に、最新の修正結果を含むものが使えれば良いという人もいるでしょう。ですので、どういう人達に、どういう結果がありますよ!、と提示してゆくのか? が知りたい所です。是非、答えが出たらお知らせ下さい!。

林：そのためには、やはり Sobalipse が、もっと使われなければ駄目だと思います。今現在は、想像のレベルで、おそらくこんな感じで使えるでしょう、でしかありません。ですので、何とか使えるように、実際の事例を増やしてゆきたいと思います。

小林：どうもありがとうございました。この会場は、もう時間が来ましたので、今日のフォーラムは、一旦、ここで終了とさせていただきます。あとは、懇親会で、是非、講師の方と、直接、情報交換をお願いします。最後に、講師の皆さん、どうもありがとうございました。(拍手)

おわりに

関西から、重い機材を抱えて、デモを交えて熱く語って下さった講師の方々、会場で積極的に発言し意見交換して下さった方々、そして、今回の企画立案から当日の司会進行までご努力頂いた小林修さん(SRA)にあつくお礼申し上げたい。

参考文献

[1] 浦下 将司 ほか, P2P フレームワーク SOBA に挑戦, UNIX USER, 2004.03, p.74-79

[2] <http://www.soba-project.org>

ムンバイの暑い風

野村 行憲

(アイシーエス)

私にとって初めてのインド出張の機会があった。短期間であり訪問先も少なかったのだが、貴重な経験と感じるところが沢山あったので、海馬の片隅で朽ちないうちに記録しておこうとしたものである。

ナマステ・ムンバイ

年度末が近づいて繁忙の兆しが出てきた2月に、以前からコンタクトのあったインドのソフトウェア企業を訪問する話が出てきた。その企業の所在地がムンバイ（かつてのボンベイ）であったため、私の最初のインドはムンバイということになった。

インドの挨拶は「ナマステ」だと思い込んでいた。大概はそれで笑顔を貰えたのだったが、お世話になった運転手の Iqbal（イクバル）さんにはナマステを拒否された。聞けばイスラム教徒なので「サラマーライクン」と挨拶するのだそうだ。次回からそのように挨拶すると、相好を崩し、これ以上無いような笑顔を見せて握手してくる。普段のいかつい顔と競技ドライバーのような運転から受ける印象とは違って、とても人懐っこい人だったのだ。

航空会社

出発の朝、成田空港の掲示板には、搭乗予定のエア・インドの便は15分繰り上げに変更されたという表示があった。繰り上がるというのが気になったので、チェックイン時に聞いてみたら、小声で「エア・インドは出発がいつも遅れるので早めに搭乗手続きを済ませるようにしている」と言う。その甲斐もむなしく、この便は出発も到着も遅れたのだった。

また、機内持ち込みの手荷物検査も厳重で、かばんを開けて中身を調べ、ボディチェックもある。ムンバイの空港を出るときの荷物チェックはX線のみだったが、担当は軍人で、彼らがボディチェックも行っていた。（彼らは親日家が多らしく、私に対して笑顔と、多分、知っている限りの日本語の単語を浴びせてくるのだった。肩に吊るした小銃を見なければ軍人であることを忘れさせる。）

エア・インドは空港の端に専用の一角に位置している。そのため売店も少なく閑散としている。

後で知人に伺って知ったのだが、日本からムンバイへ行く場合はシンガポール航空が良い選択らしい。

すべしやるでいなあ

経由地であるタイのバンコクで降りた人たちに代わって、10人位の白装束集団（とはいってもバナ〜ではない）は、どうやら何かの宗教の巡礼者のようであった。（後に彼らのことをムンバイの人に聞くと、ジャイナ教徒らしい）

離陸後まもなくして彼らのところに食べ物が配られた。それが、ちょっと異質なのである。



客室乗務員に聞くと、それは彼らが事前にオーダーしていたスペシャル・ディナーであるとのこと。珍しいので写真を撮らせてもらった。写真左から、小さいパン、バナナ、バター、紙パックのココア、小さなオレンジ、それぞれ一個ずつである。私は何でも食べられる幸いを噛みしめたのであった。

彼らの殆どは客室乗務員が話す英語もヒンドウ語も解らないようであった。

暑い歓迎

ムンバイのベストシーズンは11月から2月であると聞いていたので、できればその時季にしたかったのだが、結局は3月中旬になってしまった。

航空会社を選ぶにあたりインドということで、何のためらいも無く国営のエア・インドを選択し

たのだが、その便は深夜11時30分がムンバイ空港到着予定であった。そして、お役所仕事(?)の運行で、30分以上遅れて日付が変わってからの到着となった。

深夜の到着なので気温が下がっているはずであったが、空港を一步外に出た途端「暑さ」が襲ってきた。何しろ出発する日の盛岡の最低気温は氷点下だったので、その差は大きい。更に、滞在中には北国で生まれ育った私にとって未曾有の体験となる、摂氏39度に達したのであった。まさしく暑い歓迎を受けたのだった。これも貴重な体験である。

企業訪問

今回の出張では、3泊3日!の滞在期間中に3社を訪問した。詳しい内容は企業秘密。 :-P

・A社

ここは、以前からコンタクトのあった会社で、独立系のソフトハウスである。40~50名の社員がいる。日本をターゲットにしている、社内で日本語研修にも力を入れている。品質管理を専任で担当する部署を設けているなど、ISO-9001や、CMMなどへの取り組みも着々と進んでいて、近々Pvt., Ltd. として会社組織の充実化が予定されている。(Pvt., Ltd. については後述)

・B社

本社は米国ワシントン州のレッドモンドにあり、ムンバイはオフショアセンターという位置づけ。創業したのは元MS社の社員ということもあって、メインの顧客はMSである。社員は数十人であるが、採用条件がソフトウェア工学を修めていることが最低の条件といい、高度な技術者を揃えていてMS-Windows及び.NETについて技術力の高さを誇っている。CMMのレベル4を達成していることが特筆される。

・C社

本社は英国マンチェスターにあり、インド側はやはりオフショアセンターという位置づけで設置されている。場所はムンバイから北へ30Km程度のニューボンベイ(Navi Mumbai)に整備された、ビジネスパークにある。現状ではCMMレベル3を実現しているが、ヨーロッパ企業の系列であることか

ら、ISO-15504への関心も高い。

日本の競争力は?

いずれの企業も海外をターゲットとしている。その武器はコストと技術力であろう。現状ではオフィスコストがインドで一番高いムンバイであってもなお、人件費や物価の安さで人月コストは十分に競争力がある。また、海外からの受託ということもあって、品質管理やプロセス改善にも関心が高い。日本から単にコストが安くなるという安直にソフト開発を発注するという事は、日本のソフト産業を衰退させる危険がある。

彼らはコスト以外の競争力も身に付けているのだ。日本が負けられないためには、何をすれば良いのだろうか?

Pvt. Ltd. について

インドの会社には〇〇 Pvt. Ltd. という社名が目立つので、気になって聞いてみた。インドにおける法人の成長段階(?)には次の4つがあるらしい

・第一: Propriete for owner

又は Partner 会社

一人又は二人以上のパートナーが出資して設立。社名の後に何も表記しない。財務諸表の公開は不要。会社の収益は出資者の収益となる。

・第二: Private 会社

二人以上が出資。海外に金銭取引を伴う支店を開設できる最低限の法人。これ以上の会社は財務諸表公開義務がある。

会社名の表記は〇〇 Pvt. Ltd. となる。

・第三: Limited 会社

4人以上が出資。

会社名の表記は〇〇 Ltd. となる。

・第四: Public Limited

又は Listed company

株式上場会社。

会社名の表記は〇〇 Ltd. で上記と名前では区別ができない。



ホテルからのインターネット接続

ここからは、ホテルからインターネットを利用してみた顛末記である。

ホテルの選択

まず、旅行会社（HISを利用した）経由で予約できるホテルを洗い出し、それらのホテルのWebページから、それぞれのホテルのインターネット利用がどのようになっているかを調べてみた。

調べたホテルでは、全てのホテルで高速インターネット接続が利用でき、時間単位の料金と24時間の料金が掲載されている。

料金設定を見ると、やはり24時間利用が割安なのだが、それでも500ルピーから850ルピーとインドの物価から考えると、かなりの高額に設定されている。

（ちなみに1ルピーが2.5円、500ルピーだと1,250円相当。ムンバイではレストランの夕食が100ルピーもあれば、十分なものが食べられる。日本ではほぼ無料で使えることを考えても、やはり高額であると言わざるを得ない。）

最終的に選択したホテルは空港や訪問先への交通の利便性で The Royal Meridian Mumbai を選択した。（部屋と設備の詳細は http://www.lemeridien.com/india/mumbai/accommodation_in1956.shtml）

このインターネット接続料金は

- ・ 1時間が250ルピー
- ・ 2時間が350ルピー
- ・ 24時間が500ルピー

で、部屋の電話はコンピュータ・データポートも利用できるとあった。

接続までの長い道のり

さて、チェックイン時にインターネットを使いたいと申し込むと、無線LAN用のPCカードを貸し出すと言う。私のiBookにはPCカードスロットが無いのと、既にAirMacカード（Apple純正の無線LANカード）が内蔵されているので、PCカードは不要であると告げた。

後ほど、インターネット接続の担当者が部屋にお邪魔して設定の手伝いをする というので、部屋で待っていた。

待っている間に、無線LANカードの設定をしたら、ホテル内の無線LANステーションを捕捉して、難なく繋がったのだが、ホテルの外には接続できなかった。

そうこうしていると、部屋の電話が鳴り、出てみると接続用のIDとパスワードを 教えると言う。とりあえず、言われるままにIDとパスワードをメモしたが、そもそも、どうやって認証するのか解らない。WEPではないようだし、無線LANカードは既にネットに繋がっていることを示している。

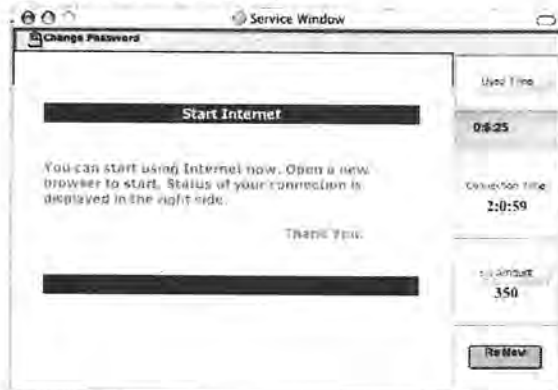
しばらく悩んでいると、部屋のドアがノックされ、インターネット接続の担当者が来た。彼はインターネット接続の注意点と、先程電話で連絡のあった、IDとパスワードが印刷された紙を持ってきた。

彼は、「ブラウザを立ち上げて、今から言うアドレスを表示してください」というのだ。そのIPア

ドレスは

「<http://192.168.100.1:45000/Login/Login.html>」というプライベートアドレスだった。(先程の紙には、このアドレスは書いていない)

そのURLを開くと、ログインのためのページが表示され、くだんのIDとパスワードを入力すると、コース選択(1時間、2時間、24時間)と、確認ボタンの画面になり、コースを選択して確認ボタンを押すと、画像のような画面が出て、インターネットに接続されたことを示す。



(こんな画面が出てくる)

試しにメールの受信などをしてみると、ちゃんと受信できた。うまく接続できたので、担当者には帰ってもらおう。

ところが...

置いていった紙を良く読んでみる。「Very nominal charges.」とあるのには苦笑したが、更に、料金表の下にこんなことが書いてあった！
Note: The billing starts from the time you login to our Network and expires on completion of the package duration you have chosen. For example; if you have purchased '1-hour package' and you login to our network at 9 pm, in spite of the fact that you didn't use the package for the complete one hour your package will expire at 10 pm.

つまり、1時間コースを選択してログインしたら、ログインした時刻からカッキリ1時間使うしか無いのだ。正味1時間をコマ切れで使うことが出来ない仕組みだ。そういえば、ログアウトするボタンが無い。

これでは、24時間コースを選択するか、インターネットは1日1回まとめて使うしかない。不便だ。

それから後で気がついたのだが、このホテルのネットワーク経由では、メールを取ってくる「POP」はちゃんとできるのだが、私が外でのメー

ル送信に使用しているSMTPサーバの踏み台チェックで蹴られてしまい、メール送信ができないのだった。(仕方なくメール送信はWebメールを使う)もちろん、SMTP以外のWeb閲覧などは問題なく使えた。

バックアップ回線の用意

色々トライしてみたが、結局のところ、チョコチョコ使うにはホテルのネットワークを使うよりは、ダイヤルアップ接続の方が、簡単で結局は安く上がるということが結論だった。こんなこともあるかと、出発前にNiftyServeの海外ローミング用電話番号(ムンバイ市)を調べていた。殆ど使わないのに毎月210円支払ってNiftyServeの会員を続けていることが、こんなところで役に立つのだ。

また、ワールドパスポートネット

(<http://www.wpn.ne.jp/>)の個人向けサービスのライトコースだと、月額基本料金が無料で、海外ローミングが利用できる。これだと接続時間に応じた従量制の料金なので、これに登録しておくといざというときに役に立つ。しかも専用の接続用のソフトがWindows版だけでなく、MacOS X版も用意されているのが嬉しい。このソフトを使うと、国、都市を指定するだけでアクセスポイントの電話番号が出てくるので、便利だ。

電話変換アダプタ

インドの電話コネクタは、俗に言うインドタイプだ。(http://mobilekit.jp/teleconne.htm参照)でも、部屋の電話機にはデータポートと呼ばれるRJ-11のジャックが付いているので、日本で使用している電話ケーブルを持っていくだけで良い。また、電話機に繋がっている部分のコネクタは大概RJ-11なので、それを抜いてモデムにつなぐこともできる。(但し、モデムは海外対応のものが良く、モデムの安全のためにモデムサーバーを経由して接続することを推奨)

電源

インドの電源電圧は240Vが一般的のようだが、ホテルの電源はWebページによると115/120Vとある。しかし、テスターを持っていかなかったので実際のところはわからない。

電源プラグはイギリスと同じラウンドピンのものだったが、良く見ると変換プラグが無しで日本の電源プラグも使えるようになっている共用コンセント

だった。



写真では変換プラグを使用しているが、空いている左側のコンセントは、日本のプラグでもそのまま差し込める。

ちなみに、iBookや最近のノートPC付属のACア

ダプタは100Vから240Vまで対応しているので安心だ。(但し、ACコード部分が100V用であることが多いので、240Vに対応しているものを用意すること)

インターネット利用は特別？

訪問先企業の人にホテルでのインターネット利用顛末を話したら、「インドではまだまだインターネットを利用するのは一般的ではなくて特別なこと。」「プロバイダの利用料金も高く、しかも速度も速くない」ということだった。

そういえば、ホテルのネットにつなぐためのログイン画面には、どこかのプロバイダの名前が出ていた。料金が低いのもそのせいかも。

海外モバイル情報

海外でのモバイルに関する情報は「地球のつながり方」<http://www.tsunagikata.com/>などの情報が参考になる。

雑感

そのほか、今回の出張で感じたことを書き留めておく。

複雑な社会

ホテルから訪問先への幹線道路から町並みや人々の暮らしの一端を垣間見ると、その貧富の激しさが感じられる。

路上生活者の屋根になっている、ボロボロのシート、敷布団代わりの古タイヤ、物乞いの人たち、スラムの住居など、とてもカメラを向けることすらできない。その一方で、高いフェンスとガードマンに守られた敷地に林立する高層マンションと、それに暮らす優雅な人たち。

様々な宗教とそれによって制限される食べ物。400を超える言語と17の公用語、そして英語を使い分ける人々。

これが私の見た人口1,300万人といわれるインド最大の商業都市ムンバイの一端だった。

出発便までの短い時間で訪れたプリンスオブウェルズ博物館では、10ルピーの入場料金が、外国人である私は300ルピー。ことほど左様に複雑である。

この複雑さを許容して、自然にながれているおお

らかさ、懐の深さがあるのだ。

南インド料理

私が今まで日本で食べていたインド料理は「北インド」のものだったらしい。それに訪問先の人たちがベジタリアンということもあって、珍しいものばかりだった。

極め付けはマクドナルドのハンバーガーで、マトンのハンバーガーやベジタリアン用のマックヴェジというのがあった。

どちらを試すか迷った揚げ句、マックヴェジを食べてみた。パンの中に挟まれていたのはカボチャコロケのようなものだった。

ムンバイ料理の代表と紹介されたのは、Panjabi Mutter Paneerという料理だ。Paneerというのは、牛乳を固めて作った豆腐のようなもので、それをサイコロ状に切ったものがメインの具で、それがトマトピューレとカレーの汁に入っている。その上にグリーンピースがあしらってる。色と形からはマーボー豆腐のようにも見えるが、肉類は入っていない。スパイスが沢山入っているので流石に辛い

さっぱりした味である。これを薄焼きでパリパリのチャパティで食べるのがムンバイスタイルだ。私がちがインド料理店で良く見る、ナーンは北インドのものであるらしい。

ヴェジタリアン

前述のように私がインドでお会いした人たちは、ヴェジタリアンなのであった。日本では精進料理くらいしか思い当たらないので、いったいどんなものを食べているのか、カロリーは十分なのか、など疑問は絶えないが、一緒に食事してみると、これが中々奥が深い。ヴェジタリアンといっても宗派によつては、土の上に成るものしか食べられない人たちも居る。この人たちはカボチャは良いが、芋はダメなのである。ヴェジタリアン向けの食品には緑色の日の丸のようなマーク(100% vege)が付いているので、識別はたやすい。

とは言え、先程の Paneer やカボチャなど結構カロリーが高いものも多く、加えてバターや油を多用するので、実際はヴォリューム満点なのである。

そのせいか、お腹がポコッと出た人が目立つ。サリ姿の女性もポツテリと弛んだ脇腹が顔を出しているのを良く目にする。カロリーは十分なのだ。

そういうわけで、インド滞在中ず〜っとヴェジタリアン食を満喫したのであるが、体重が減ることは期待できない。(Vegeは成長の意味もあったな)

そうそう、更に彼らはお酒もタバコも吸わないのだった。どちらもダメな私は、最初は気付かなかつたが、お酒を勧められることも無く、副流煙に煙ることも無くて、とっても楽だったのだ。

オートリキシャは凄い

私には石の下に集まっているダンゴ虫に見えたほど、沢山のオートリキシャが、それぞれウジャツと走っている。凄い。この凄さは私の稚拙な筆力ではとうてい表せない。幹線道路であっても車線という概念は無いらしい。ちょっと隙間があるとドンドン入ってくる。2〜5台が並走する中を、人々が歩いたり横断している。これもこの国の社会と同様に渾沌とした秩序(?)で流れている。

しかしオートリキシャの台数はものすごい。いっぱい走っているほかに、道端に直角にぎっしり詰まって駐車している。ドアが無いのでピッタリくっついていても乗降できるのだ。

オートリキシャは小さな三輪車で、屋根が付いていて、後ろの座席には、無理をすると大人が三人位座れるようになっている。規格化されていて屋根が

黄色で、ボディは黒く塗られている。後はオーナーの趣味で色々な飾りがついていたりする。

町の名前を失念したが、ニューボンベイ(Navi Mumbai)への途中にある町はオートリキシャが入れない地域で、そこでは少し大型のオートリキシャのようなものが走っている。

オートリキシャの他には、ちゃんとドアの付いた軽自動車のようなタクシー(キャブ)が走っている。ムンバイのシティと呼ばれる中央官庁街のあたりは、オートリキシャの乗り入れが禁止されているので、そこではキャブが幅を利かせている。幅を利かせるとは言っても、車幅は小さいのだ。

通勤電車の混雑は東京以上

ムンバイは電車が発達していて、通勤には便利であるらしい。とは言うものの、電車の混雑は東京以上らしい。電車の客車を見ると、出入り口が大きいのだが、扉は無い。加えて窓は付いているものの小さい窓が、ほんの少しついている程度だ。乗客はフレッシュエアを求めて、扉の無い出入り口にタムロしている。ラッシュアワーには、鈴なりにへばりついているとのこと。「日本人は乗れないでしょう」と言われたが、私も乗ろうなどと思わない。

振り落とさないようにという配慮か、走行スピードはあまり早くないように見える。

洗濯はただけない

インドではクリーニングに出さないほうが良いとは聞いていたが、試しにワイシャツ1枚とアンダーシャツを1枚を出してみた。

夕方部屋に戻ってみるとワイシャツはプレスされて、ビニール袋に入ったままつるされている。漂白剤で青くなっているわけではない。以外と良いのでは?と思ったが、襟を見てがく然とした。襟の真後ろの部分に、あきらかに洗濯挟みで吊るされた跡がくっきりと付いている。これは自宅に戻ってから数回洗濯とプレスを繰り返したがまだ消えない。

アンダーシャツは何と箱に入れて置いてあった。期待してフタを開くと、更に柔らかい紙でくるまれていた。がっ! とってもヨレヨレになって変わり果てていた。

結構高級なホテルだったはずだが、インドのクリーニングはおそろべしなのだ。

ともあれ、異文化体験は面白い!

文責:野村 行憲(ICS)

紺屋の白袴

SIGENV Member 有志

SEA の IP までの道のり (2003年・夏)

1. はじめに

まずいことに(?), あの阪神タイガースが優勝しそうである。いや、昨年も5月の時点ではそうであった。しかし、今年こそ本物のようである。「あの黄色い SEAMAIL」と話してわかる人も少なくなってしまった。

SEA 設立当初の1年間は、「阪神タイガースの優勝を記念して、機関誌の表紙の色は黄色にしました。以後、発刊年毎に色は変えます」と、機械振興会館の会議室で開かれた設立総会で、発起人の一人で、SEAMAIL 編集長の岸田さんが挨拶されたことを思い出す。外は寒かったが、何か新しいことがゆっくりと起きそうなことを予感させる、熱気に満ちた12月の夜だった。

[編集部注] ほんとうに阪神が優勝したので SEAMAIL の表紙は20年ぶりにまた黄色に戻りました。

しかし「監督が変わると選手はこうも変わるものなのか」と思う。世代という意味の語 generation を、辞書で引くと40年と書いてある。それからすると、SEAの歴史はその半分も経過していない。が、呑み屋も3年も持てば一丁前というから、十分に歴史があるのかもしれない。消え去ってしまうことが実は歴史の大部分なのだとすることを、痛感するこの頃である。記憶があまり不鮮明にならないうちに、SEAの事務所やマシンの遍歴の記録も兼ねて今回のIP化までの道のりを書き留めておこうと思う。

2. 東高円寺～隼町：ネットワークの黎明期に辿り着くまで

1985年は、色々あった年だった。まず、予想外にタイガースが優勝してしまった。そして、産業界では、ビックブルーといわれたIBMの大型汎用機の断然の優位は雲行きがあやしくなっていた。一例は、日立のIBMスパイ事件だったりした。しかし、まだ所々にしか雲は見え、また新たな胎動に、目を向けようとしていた。AIブームの先駆けとなる第5世代もちょうどこの頃だった。

当時、何が衝撃的だったか?。個人的には、SEAMAIL という印刷物が、当時としては高水準の印刷物だったことである。創立当時の会員数は、高々150人である。倍の数を作ったとして300冊。とても信じられなかった。その頃の印刷物は、岸田さん(SRA)を中心に、ボランティアベースで、日本語化した roff とその亜種(nroff, ditroff)を使って版下を作成していた。会員管理は、野辺良一さん(SRA)が、東高円寺のマンションの1室を借りた仮事務所で、NEC PC9801VM(MS-DOS)上の informix で作成していた。パソコン通信が一部のパソコンユーザに浸透し始めた頃の話である。

設立してから1周年を過ぎたあたりで、事務所が手狭になったことや、駅から近いとはいえ、やはり多くの人からアクセスには至便とはいえないことから、関係者の多くが勤めている平河町の近くということで、隼町のマンションの一室に移転した。事務局から貰った名刺の裏には87-10-28という日付が書いてある。この頃になるとやはり会員も増えてきたので、UNIX-WSで会員管理をとということもあり、SONY NEWSを購入して informix をMS-DOSから移植するという仕事もあった。こころあたりは、田中慎一郎さん(SRA)が尽力された。

また、しばらくして、平河町のSRAの一室をJUSの事務局と一緒に間借りさせて貰ったあたりで、junetへの接続がされたように思う。

3. junet からの商用接続へ

色々な意味で、junet と関連があった人が多い関係で、SEA は、早い段階で SRA 経由で junet に接続していた。もちろん UUCP 経由である。そのために、アドレス sea.or.jp を取得した。そのあと、数年後に、バブルが弾けはじめた1990年の夏に、事務所もいまの四谷三丁目に移った。

しばらくして、商用のインターネットが始まり、当時、代表幹事だった中野先生から「SEA も商用ネットワークと正式に繋ぐのが適当」というアドバイスを受け、幹事会で了承されたのが1994年頃である。以来、28.8Kbps のアナログモデムで UUCP で接続していた。せっかくの機会なので、そのあたりのマシン環境を簡単に紹介しておこう。

3.1 SONY NEWS

SEA が誕生したころの、ソフトウェア業界の大きな話題は、なんといっても、あのシグマ計画だった。私などは、そんなことは微塵も関係ない職場にいたので特に仕事上で、直接、これらのプロジェクトについて関与することは無かったが、Unix に縁のあった当時のソフトウェア技術者の多くは、それに関しての話題が少なからずあったと思う。まあ、そんなことはともかく、SEA にあった NEC PC-9801VM の後継に選ばれたのが、SONY NEWS である。4.3BSD に独自の日本語拡張ををしたマシンで、HD も SCSI のものが 20MB くらい付いていただろうか。SEA の業務の上では、このマシンに、Informix RDB を入れて会員管理 DB が動き出した。

3.2 X 端末の導入

「ワークステーションが技術者に1台ある理想の環境」を羨んでいた人も多い中、SEA では、四谷の事務所で業務を始めてしばらくしてから、カラーの X 端末を2台導入した。ディスプレイには、カラーの大きな17インチくらいのものを装着した。

ディスプレイは、その後、TFT 液晶に買換えたりしたが、いまでもかろうじて現役で稼働している。壊れず元気に稼働しているので、捨てるに忍びないが、近日中に退役させる予定でいる。

3.3 SONY-NEWS の延命・退役と DOS/V PC + FreeBSD

SONY-NEWS の老化は、頭痛の種だった。不要になった同型機の部品を分けて貰ったりして、何度か危機的な状況を脱してきたが、それもその場凌ぎのことでしかないのは明白だった。一番頭が痛かったのは、当時の機材は、メーカーの独自仕様のもので多く、特に HD の容量確保（増設）がままならなかったことである。当時の SCSI は、互換性に乏しかった。

3.4 DOS/V PC + FreeBSD

PC で Unix という、いまでは当たり前だが、昔は、やはり一部のマニアックな人たちのおもちゃでしかなかったような気がする。秋葉原に行っても、一部のマニア向けの店の隅に、ひっそりと「NetBSD, FreeBSD」といった CD-ROM が売られていた。そんな折、環境分科会で、実践的に分科会活動をするために、DBMS などの周辺をマシンを使ってやりたいね、という話が出て、PC の構築をサポートしてもらうことになり、機材を調達した。

Pentium 166MHz、メモリ 64MB、HD 2GB といったパーツを、秋葉原に出かけて買ってきて組み立てたのが1997年5月末の、大雨の1日だった。FreeBSD 2.2R をインストールしたら、あっけなく作業が済んでしまった記憶がある。

それからしばらくして、SONY NEWS の調子が本格的に悪くなり、結果として、このマシンを SEA の中核マシンとして供出することになる。その後、今年(2003年)、退役させるまで、HD や NIC の障害は何度か発生したものの、それ以外の障害はなんら無く、無事安定して運用してきた。

3.5 Informix から Oracle へ

当初、環境分科会で、共通問題を考えるときにSEA-DBは、PostgreSQLなどのオープンソースのもので作ろうと考えていたが、皆さん、多忙なため、なかなか実作業に着手できないうちに、上記のような理由で、マシンを供出したこともあって、遅々として進まなかった。しかし、SEAの基幹のシステムでもあるので、いつまでも動きませんということでは支障があるので、伊藤さん (NilSoft)にご協力頂いて、Windows PC に OracleLite でシステムを作って頂いた。いまでも元気に動いている。

3.6 sea.jp の取得とIP接続

2000年の暮れくらいだったと思うが、汎用jpドメインが公開された。これに関して、旧来からドメインを取得しているものに優先的に割り当てられるものがあり、これを機に sea.jp を取得した。このドメイン取得の引換えに sea.jp を IP 接続する必要があるので、それを実施するなら、当然ながら、SEAの事務所をIP接続しましょう、という決定が幹事会でされたのが2001年の1月頃。仕事の片手間で、準備をして、そのうちやりますと2001年度のSEAの総会で話をしてから、sea.jp を、IP接続したが、従来からの sea.or.jp などの接続環境は、従前通りであった。

4. 現在の SEA のマシン環境

世の中、ADSLがあつという間に普及して、従来では考えられない大きさのメールが行き交うようになると、さすがにUUCPでの接続では耐えられなくなってきた。そんなこともあって、本格的に sea.or.jp も含めた SEAのIP接続移行を考えて動き出したのが、2002年の秋である。それからボランティア作業で、暇を見つけて仲間が集まって作業を進めてきて、メールの移行などもすませた。

せっかくなので、現在のSEAの、特にネットワークの構成・環境については、別稿で、簡単に紹介したい。

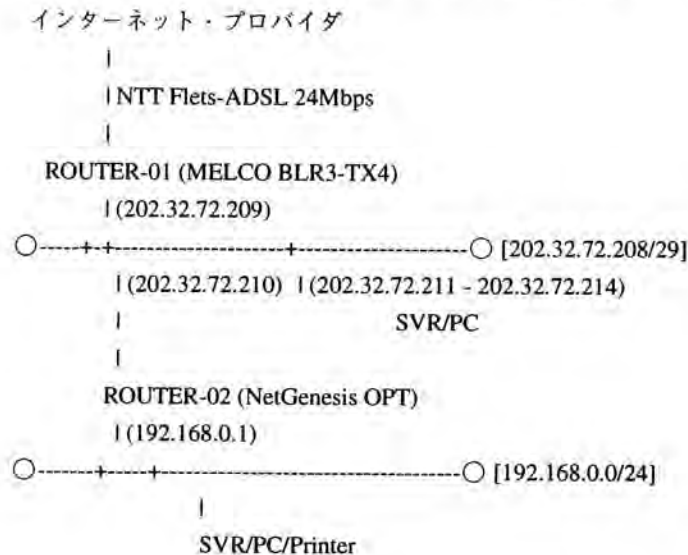
5. 今後のことなど

せっかく安定したIP環境が構築できたので、やはりSEAの活動の一助になって欲しいと考えている。たとえば、SEAを理解してもらおうという点では、いままである実績・記録を、整理して文書や画像、音声などを公開していくことも必要であろう。また、シンポジウムや分科会活動などのリアルタイム中継やオンライン開催なども試行されてよいのかもしれない。

随分前に、書きかけた原稿に加筆修正している。まあ、それくらい遅々として進まない作業を、脇で見守って下さった、岸田事務局長、深瀬・前代表幹事ほか関係者に改めて、御礼申し上げたい。

SEA のネットワーク構成について (2003年・夏)

前稿で紹介したように、SEAのIP化作業が一段落したので、現時点(2003年8月)のSEA事務所内のネットワークの構成(主要機材、経路設定、メール、Webのサービスの状況)を簡単に紹介する。ネットワークのトポロジは、次の図のようにきわめてシンプルなものである。



A. 主な構成要素

A-1) インターネットとの接続

廉価な ADSL 回線を使っている。当初 1.5Mbps で接続をしていたが、SEAの対外接続のIP移行がされたので、2003年8月に ADSL 24Mbps に移行の予定である。ADSL ルータは、2代目である。初代のものは、同じメーカーの旧型機種であったが、1度の故障修理と2度の不安定な動作により、退役させた。

A-2) DNS

外向き DNS は、現在、プロバイダのサーバでサービスしている。内向き DNS は、202.32.72.208/29 上のマシン(FreeBSD)に dnscache/tinydns で稼働させている。

A-3) メールサービス

202.32.72.208/29 上のマシンに MX を向けてある。このマシンがリレーして内側の外向き DNS は、現在、プロバイダのサーバでサービスしている。MTA は、FreeBSD 上で qmail を使用している。

A-4) WEBサービス

202.32.72.208/29 上のマシンで www.sea.jp, www.sea.jp をサービスしている。サービスには apache を使用している。

A-5) http-proxy

192.168.0.0/16 からインターネットの利用するために、202.32.72.208/29 上のマシンで http-proxy など内向けにサービスしている。サービスには squid を使用している。

B. 経路設定

B-1) ROUTER-01 (MELCO BLR3-TX4)

インターネットプロバイダとの接続を行っている。デフォルトルートを、WAN側に向けている。このルータは、内側の（プライベートアドレスの）経路を設定していない。

B-2) ROUTER-02 (NetGenesis OPT)

インターネットとプライベートのローカル接続を行っている。このルータは、夫々の足元に接続してあるネットワーク（及びプライベートアドレス用のIP帯域をブラックホールに向けている）以外は、どこにも経路を設定していない。

B-3) 202.32.72.208/29 上のマシン

原則として、プライベートアドレス向けを ROUTER-02 に向けて、デフォルトルートを ROUTER-01 に向けている。

B-4) 192.168.0.0/16 上のマシン

デフォルト・ルートを ROUTER-02 に向けている。

C. 運用状況**C-1) メール**

約 800通/日のメールの受信がある。平均すると約2分に1通のメールを受信していることになる。IP化以前は、28Kbpsのアナログモデム経由のUUCP接続だったので送受信に相当の時間と費用を要していたが、いまは、ほとんど、ストレス無くメールを処理している。

また、IP化を機に、MTAを従来の sendmail から qmail に移行し、不正ID宛でのメールも全てゴミ箱に入れて、無用な user-unknown などの返信処理をしないようにした。

MXホストの2重化は、年内には実施する予定でいる。停電などを考慮した（SEAの事務所以外での）MXの2重化などは、今後の課題としたい。

C-2) Web

従来は、プロバイダのサーバ上と、SEA事務局内のサービスを併行に稼働させていた。しかし、マシンや回線が安定すれば、このまま事務局内でのサービスに一本化していく予定でいる。

利用状況などは、アクセスログは収集しているが、きちんとした分析などは行えていない。

特に、アタックと思われるものもログに多数記録されているので、今後、それらを元に、不正アクセスの対処を機敏に行うための仕掛けを準備していきたい。

おわりに

SEA事務局内のネットワークの構成を簡単に紹介した。ボランティアベースでのSEAの活動は、このようなネットワーク整備1つとっても、やはり一度にできることは限られている。しかし、ソフトウェア技術者の基本能力として、ネットワーク技術の習得と実践は、現代の必須事項だと思われるので、それに追従すべく、会員諸兄のご支援をよろしく願いたい。

<p>その後の経過 2003年・秋～2004年・春</p>

A. 2003年9月・幹事会での報告

最近行ったことは次の通り。

A-1. DNS の変更

sea.or.jp, sea.jp を WEB 用に設定しました。いずれも同じ IP を Aレコードとして持ちます。それに対して www.sea.or.jp, www.sea.jp を CNAME として定義してあります。

各位へのお願い：今後 SEA の Web ページは、www.sea.jp ということでお願いします。ただし、上記に述べたように実態は同じです。

また、近いうちに、従来から公開している //www.ijnet.or.jp/sea/ には、自動転送のスク립トを挿入し、年末くらいには閉鎖したいと思います。

[編集部注] 現在すでに //www.ijnet.or.jp/sea/ は閉鎖されました。

A-2. 対外サーバの冗長化

特に、メールのロストを防ぐために、OS などのバージョンアップなどを安全に進めるためにも、マシンを2台体制にするべく最終的な作業を進めています。

今後の予定は次の通り(プライオリティの高い順):

1. merman.sea.jp (外向けサービスマシンの1台)のリプレイス: FreeBSD 4.9R のリリース待ち。
2. marmaid.sea.or.jp (事務局内部使用)のリプレイス: Win98 で SEADB を現状のまま移行する。これで数年延命!
3. halcyon.sea.or.jp (内向けサーバ)のリプレイス: FreeBSD 4.9R のリリース待ち。groff 環境に関しては、現状マシンと併行して運用するかも構成これから検討するが、RAID も検討したい。
4. テスト用のマシンの準備: swiki や新サービスなどの実験環境。先日来、幹事会 ML で提案いただいた swiki などの Web の実験環境は、本番環境とは別に、できるだけ早い時期に構築できればと考えています。
5. MLサービスの整備: 運用労力の軽減、緊急時の対応などを検討。

B. 2003年10月・幹事会での報告

- 1) FreeBSD 4.9R のリリースが遅れているので、大きな進展はなし。
- 2) 運用中のプログラムのパラメタなどの微調整、不具合修正などを実施。(sendmail の qmail の互換モジュールとの置換、squid のパーミッションの不具合の修正など)。

C. 2004年2月幹事会での報告**C-1. インターネット接続用ルータの不具合**

1/2 AM 1:46 - 1/5 9:57 に発生。ルータの再起動で復旧以後、トラブルは発生していない。現運用機は、昨年7月に購入したものであるが、Dos 対応を実装し、廉価で構成な後継機が発売されているので、近いうちにリプレイスして、現運用機は予備機に充てる予定(購入済み: 5,400円)。現予備機は、性能的にも劣るので廃棄する。

C-2. marmaid のハードウェアリプレイス(12月実施)

事務局内部で使用している Win98 機のハードウェア交換。一部不安定なところもあるので、後日

再調査、再調整する。

C-3. 今後の作業予定

halcyon (ローカル側のメールなどのサーバ) のリプレイス: FreeBSD 3.3R --> 5.2 R へ, ハードも含めて, リプレイス j-groff の不具合は継続調査中. 3月末くらいまでに新マシンを仕立てて, しばらく旧マシンと併行稼働させる. 新マシンでの追加サービスは, DHCP, proxy-pac を予定. これで, SEA 会員が Note PC などを事務局に持込んでも, 容易に作業ができると思う. これとは別に, マシン (旧マシンを再構成するか新規に購入するかは後日検討) で, swiki, IMAP, IPv6, 新ML などを順次テストしていきたいが, ...

D. 2004年4月幹事会での報告

2月から3月にかけてのSEA事務局マシンの整備状況は以下の通り.

2月5日

DNS の切替: メール对外サービスのホストを1台のみにした. マシンが安定稼働していること, および, 実験用のマシンとして1台を割当てたいため.

3月1日 AM01:21

内部メールサーバの reboot: 原因不明. 調査中だが, 情報不足で終りそう. 外部からのアタックではないと思いたい. 現在, マシンは安定稼働中. なお, 先月内部サーバのリプレイスをしたと報告したが, その後関係者で検討をした結果, 各種ソフトの移行の手間などを考慮し, 現体制をもう少し継続することにした. 新マシンで予定していたサービスのうち, DHCP や proxy-pac のサービスは現マシンで稼働開始(2/4).

3月3日

- 1) III 向けルータのリプレイス: セキュリティ上の問題があったので, 一旦, 元の機材に戻すセキュリティ対策を万全にしたのち, 再度, 本番リプレイスを計画する.
- 2) FreeBSD 用の緊急パッチの適用 (Security Advisory SA-04:tcp)

3月4日

新ルータ交換不具合 (インターネット側から管理画面がえてしまう) ので, しばらく旧ルータで運用することにした.

3月中旬～

SPAM が大量に到来しだしたので, 着信拒否などの対応をする. また, 不正な宛名向けの着信メールを毎日レポートするためのスクリプトなどの整備に着手.

Preliminary Call for Papers

ISFST-2004

International Symposium on Future Software Technology 2004

October 20(Wed) - 22(Fri), 2004 in Xi'an

Co-sponsored by

Software Engineers Association of Japan
UNU International Institute for Software Technology
Northwest University
Xi'an Software Engineering Institute

ISFST-2004 is the 8th International Symposium on Future Software Technology.

The previous seven symposia in this series were successfully held in China since 1996. The series have been co-organized by SEA (Software Engineers Association) of Japan, UNU/IIST (International Institute of Software Technology of United Nations University), and the local host academic organizations.

This year, Northwest University and Xi'an Software Engineering Institute will be our host.

The purpose of the Symposium is to review the recent progresses in software technology, to exchange various technical and business information, and to promote the sound industry-academia cooperation for future. It will provide a unique opportunity for practitioners, researchers and educators in software community to gather latest information and discuss the advances and the trends in the field. Program committee is looking forward to receive your submission of research papers, experience reports, and tool demonstrations related to the following topics.

Topics: Suggested topics include, but are not limited to:

Engineering:

- System/Software Development Paradigm (e.g.: Open Source)
- System/Software Evolution and Maintenance
- Database/Knowledge/Data Mining, Discovery and Engineering
- Requirement, Product Line Engineering
- User-Interface, Architecture, Testing
- Methodologies, Modeling, Framework

Internet:

- Web-based System/Software Engineering
- Security, Web-based system (e.g.: Web services)
- Mobile and Multi Agent system

Management:

- Process, Project, Testing and Quality Improvement/Management
- Human Resource Development/Management
- Distributed/Concurrent/Cooperative Software Development

Theory:

- Formal Methods, Algorithms, Logics, Fault Tolerance, Reliability
- Software Evaluation, Verification and Validation, Quality Assurance
- Communication, Multimedia, Graphics, Simulation, Languages
- Parallel/Distributed/Realtime computing

IT in general:

- Industry Experiences and Survey
- New Application Technologies and Domains (e.g.: eCommerce)
- Business Process Re-engineering, Cooperative Work Support, Education
- Object-Oriented Analysis/Design/Programming/Testing, Embedded Systems

The unique feature of the ISFST symposia is that the program includes informal but intensive working group style discussion sessions besides ordinary-style paper presentations. Industry people are encouraged to propose specific discussion topics, demonstration of tools and/or report their experience instead of submitting regular papers.

Xi'an is the one of the oldest cities and have unique historical background and culture. As well as being an ancient walled city, Xi'an is famous for the "terracotta warriors", created over 2000 years ago for the mausoleum of Emperor Qin Shi Huang. All the attendees from other regions of China and visitors from outside China can enjoy their stay at Xi'an during the symposium.

Paper Submission:

Submissions for a full paper or an extended abstract should be received no later than June 30, 2004 by the following address:

Email_Only_to: isfst2004-paper@media.osaka-cu.ac.jp

Paper submissions should contain title, author names, affiliation and address, abstract, list of keywords and applying category of topics followed by the text within 5000 words in 6 pages.

Experience report submissions should contain title, author names and affiliation and address, abstract, list of keywords and applying category of topics followed by the text within 2000 words in 4 pages.

Discussion topic proposals for working group sessions should contain title, proposer names, affiliation and address, list of issues to be discussed followed by proposer's position statement within 2 pages.

Tool demo submissions should contain the name of the tool, author names and their affiliation and addresses, list of keywords, followed by the text describe tool function within 2 pages.

Only electronic submission in PDF, portable Postscript or Word format via e-mail will be accepted.

Note: The paper should be written in English only. Helvetica (or Arial) and Time (New) Roman fonts only.

Use local fonts free author's kit at:

<http://www.sea.jp/Events/isfst/ISFST2004/>

Important Dates:

Submission deadline:	June 27 (Sun), 2004
Notification of acceptance:	August 22 (Sun), 2004
Final paper due:	September 19 (Sun), 2004

Paper selection will be based on originality and contribution to the topics. Papers are reviewed formally by more than two PC members and will be judged at the PC meeting on early August.

SEA, Japan will publish the symposium proceedings with ISBN number in CDROM after the symposium by compiling the papers accepted and presented at the symposium and the summary report of working group discussions, etc.

ISFST-2004 Staffs**General Co-Chairs:**

Kegang HAO (Northwest University)
Akira KUMAGAI (TELST)

Program Co-Chairs:

Zhi JIN (Chinese Academy of Science)
SooYong PARK (Sogang University)
Kazunori SHIOYA (SRA-KTL)

Program Committee Members:

To Be Selected Soon.

Web Publicity Chair and Mailing list manager:

Hideo NAKANO (Osaka City University)

Advisory Staff:

Keijiro ARAKI (Kyushu University)
Dookwon BAIK (Korea University)
Chris GEORGE (UNU-IIST)
Dehua JU (ASTI Shanghai)
Xichang ZHONG (SEC-CAS)

Local Arrangement:

Wei GE (Northwest University)
Xiaoqing GONG (Northwest University)

Treasurer:

Kouichi KISHIDA (SEA)

Official Web Page:

<http://www.sea.jp/Events/isfst/ISFST2004/>

Web Page inside China (Coming up soon):

<http://www.nwu.edu.cn/page/xueshuyuandi/isfst/>

編集後記

☆

4月初めに PROFES 2004 のチュートリアル講師として来日された韓国 POSTECH (浦項工科大学) の Kang 先生をお招きして、大阪で特別 Forum を開催しましたが、そのときに使われたスライドを、先生のお許しを得て巻頭に掲載しました。貴重な情報が詰まっていますので、ご活用ください。

☆☆

Product Line は最近あちこちで注目を集めている話題で、大阪の Forum もかなり盛り上がりました。今年の12月初めに APSEC (アジア太平洋ソフトウェア工学会議) が韓国・釜山で予定されていますが、その Co-Located Event の1つとして、Product Line Software Engineering についてのワークショップを開催しようという計画が Kang 先生を中心に進められています。いずれ開催が決まりましたら、あらためてご案内します。

☆☆☆

2月中旬に開かれた SOBA フォーラムの記録を、編集ボランティアとして、渡邊雄一さんがまとめてくださいました。各支部や分科会での集まりについても、こうした記録をどなたかが報告していただくと、編集部としてはたいへん嬉しいのですが、.....

☆☆☆☆

野村行憲さんからは、インド出張の旅行記を寄稿していただきました。SEA 会員の中には、開発アウトソーシングに関係しておられる方々も多いと思います。それぞれの体験を発表していただくというのも、有意義な情報交換だと思うのですが、.....

☆☆☆☆☆

「紺屋の白袴」と題された巻末の文章は、SEA 事務局のマシンおよびネットワーク環境整備をボランティアしていただいている SIGENV (環境分科会) からの状況報告です。

☆☆☆☆☆☆

いずれはこの SEAMAIL も、ハードコピーではなく Web Magazine に変わることになる日がくるでしょう。さて、いつのことか？

☆☆☆☆☆☆☆

岸田孝一 @ SEA Office



ソフトウェア技術者協会

〒160 東京都新宿区四谷3-12 丸正ビル5F
TEL.03-3356-1077 FAX.03-3356-1072