

Challenges in Software Engineering in Japan

- ❖ Nationwide Software Disaster
- ❖ Embedded Software Crisis
- ❖ Challenges and Our Strategies

Mikio Aoyama

Nanzan University, Japan

mikio.aoyama@nifty.com

<http://www.nise.org>

ICSE 2004, May 26, 2004, Edinburgh

**NANZAN
UNIVERSITY**

NISE: Network, Information, and Software Engineering Laboratory

All Rights Reserved, Copyright Mikio Aoyama, 2004

Nationwide Software Disaster Banking System

➡ **April 1, 2002: The Banking System of the Largest Bank, Mizuho FG (Financial Group), Went in Services but Malfunctioned for 1 Month**

➡ **Mizuho FG is the Merger of 3 Major Banks in Japan**

➡ **The 3 Year Project of Bridging 3 Major Banking Systems of Different Computers/OS/Applications Failed**

✋ **Internal Political War Made the Project Turmoil**

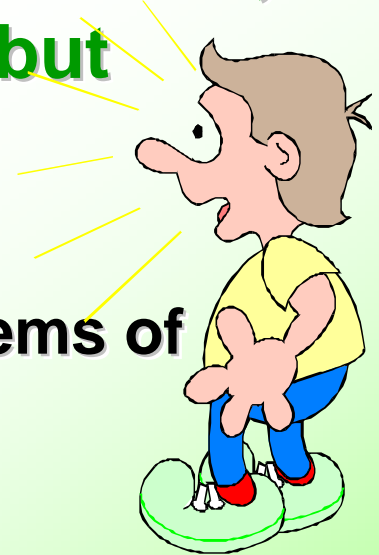
✋ **Without Completing Testing, The System Went into Services**

➡ **Disaster in Daily Life: Thousands of ATM Were either Out-of-Order or Malfunctions**

✋ **Double Withdraw from 60,000 Accounts**

✋ **Delay of 2.5 Millions Transactions**

➡ **People and Government Realized the Development and *Operational* Risk of Software**



Nationwide Software Disaster Transportation System

☞ **March 1, 2003:** A Bug of FDP (Flight Data Processing system) Made Tokyo ATC (Air Traffic Control center) Almost Shuttered for Whole Day

☞ A Revision of FDP Exposed a Hidden, Known and Unfixed Bug

☞ 203 Flights Cancelled & 1,500 Flights Delayed upto 6.5 Hours

☞ Affected 270,000 People, 1,000 People Forced to Stay in Night at Tokyo Airport



☞ **April 8, 2004:** A Bug of RDP (Radar Data Processing system) Triggered FDP System in Tokyo ATC Down for More than 2 Hours

☞ 130 Flights Delayed More than 30 Min.

➔ **Software is Social Infrastructure**
Software is Vulnerable, So the Society is

Airplanes Grounded on March 1, 2003

Nationwide Software Disaster Mobile Phones

☞ **Feb. 2001: A Bug of Embedded Software Forced 230,000 Recall of Mobile Phones, Damaged \$Mill Loss**

☞ **Increasing the Size and Complexity of Embedded Software**

☞ **Other Companies Repeated Recalls or Stop Selling**

☞ **People so Depending at their Personal Life**

☞ **Camera, Game, Digital TV & Radio Receiver**

☞ **e-ID and Digital Wallet: (Like IC Card) Check-In/Out JR Train and Domestic Flights, e-Cashing**



➔ **Risk of Embedded Software is Huge to both Customer and Vendor**

Embedded Software May Cause Recall of Systems
3rd Wave of Software Crisis? (after Mainframe & PC)

Embedded Software Crisis

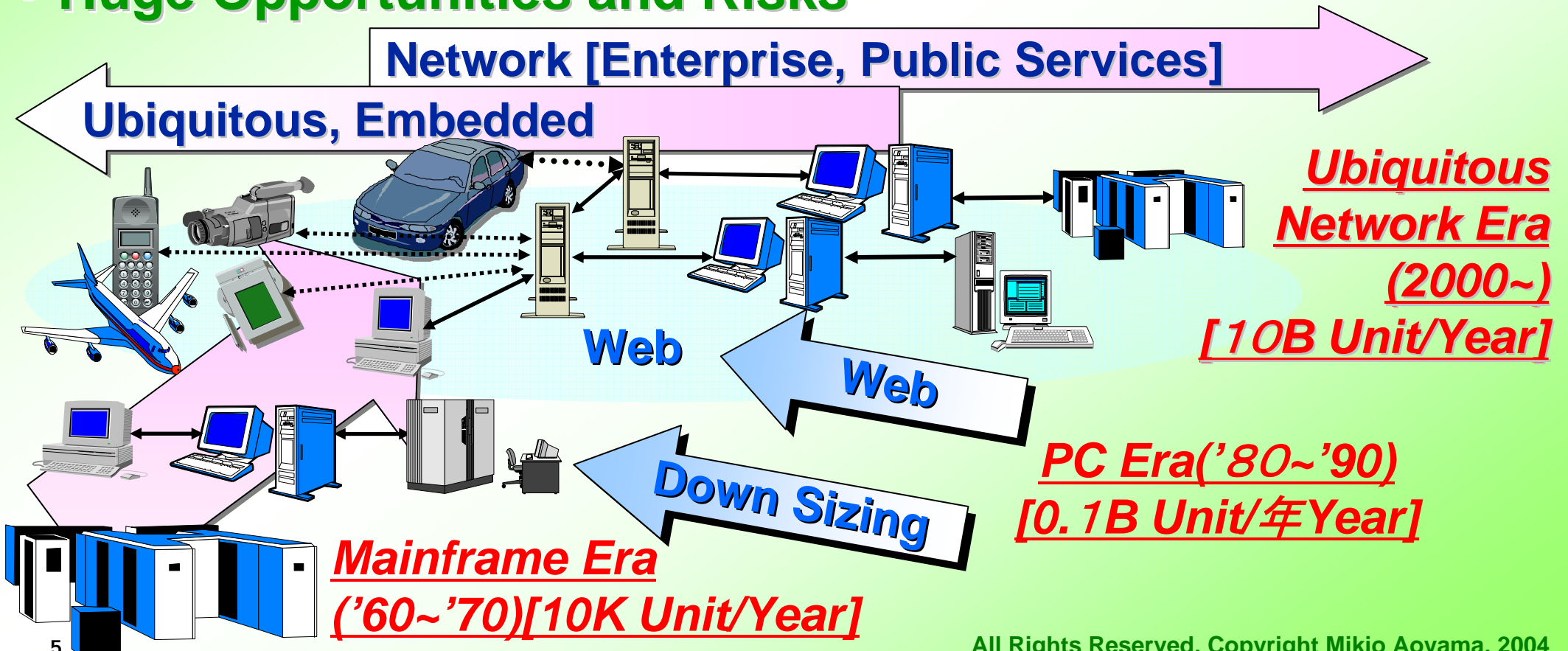
From PC to Ubiquitous/Embedded Network

➤ Another Turning Point: 3rd Wave of IT Evolution

➤ From PC(0.1B Unit/Year) to Embedded Controllers (10B Unit/Year)

➤ Ubiquitously Connected: Dense Interaction and Autonomic

➤ Huge Opportunities and Risks



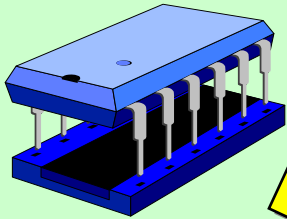
Embedded Software Crisis

Repeated Crisis from Mainframe, PC to Embedded

wavering
Path?

- 👉 Huge Gap between Demands and Supply
- 👉 Reinventing Software Engineering
 - 👆 In Early Stage of Software Engineering
- 👉 Much Enthusiasm and Expectations

Conventional
Embedded
Software



Software is
Addendum
to Hardware

Functionality

Size & Complexity: 100K \Rightarrow M LOC

Productivity and Cost

From Experience to Engineering

Openness

From Inhouse Vertical Integration to
Horizontal Integration

Integration Complexity

From Single System to Networked Systems

Social Impact

Everywhere Software

Strategic Change

Safety and Reliability as Infrastructure

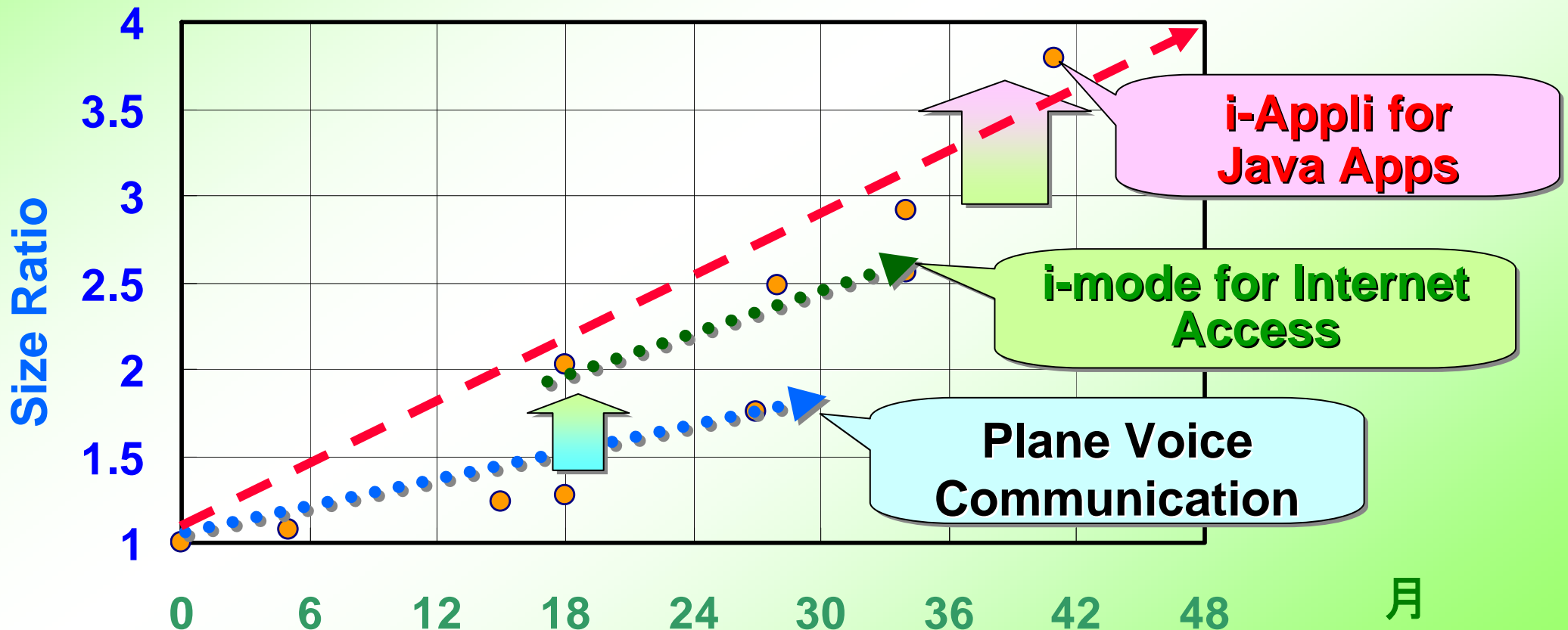
Embedded Software Crisis

Evolution of Mobile Phone Software

👉 X4 Evolution of Software Size for 4 Years

👉 Reaching to Large-Scale Software Systems: ~2M LOC

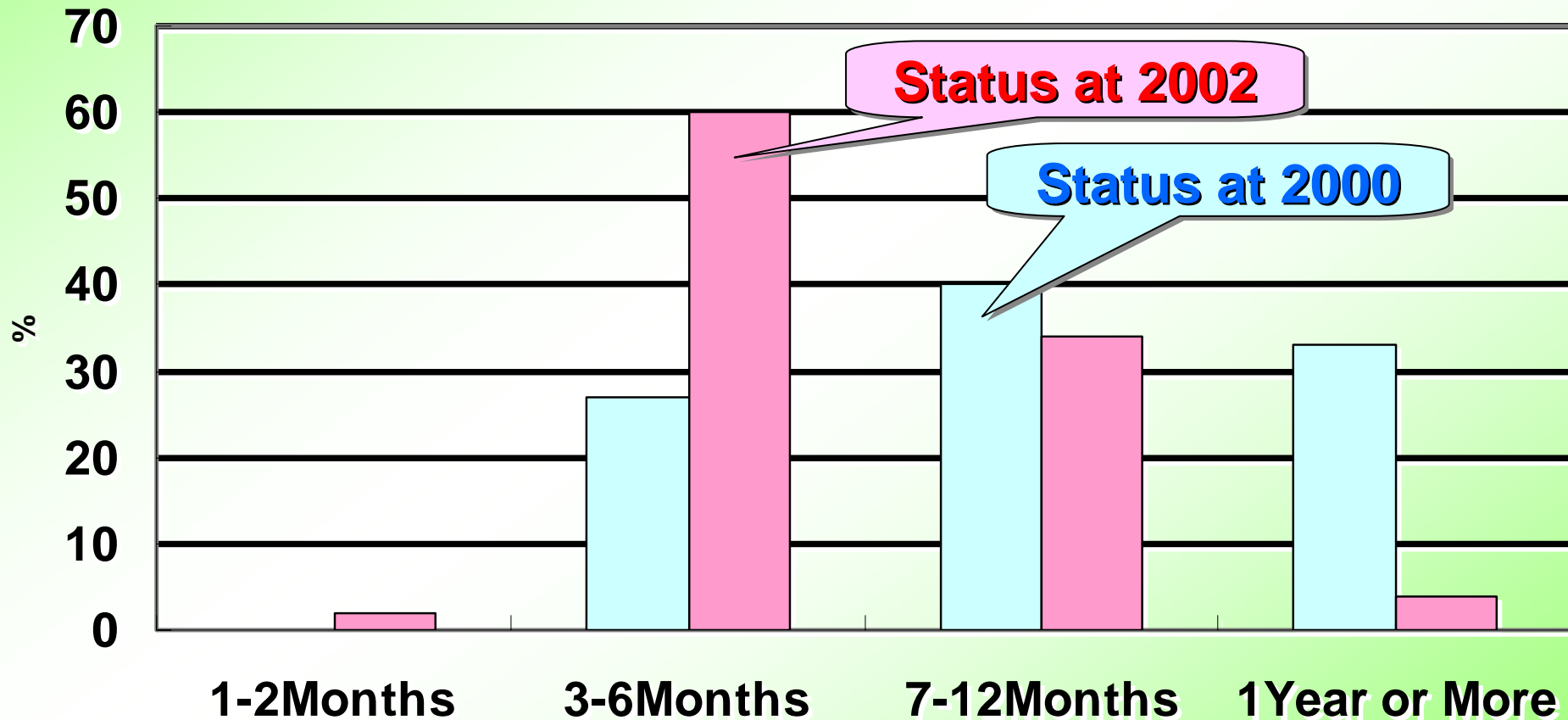
👆 Mobile Phone, Digital TV, Car Navigation System



Embedded Software Crisis

Shorter Development Time is Norm

- 👉 60% of Projects: 3-6 Months for Development
- 👉 Time-to-Market is Competitive Edge



Survey by Mitsubishi Research Institute for IPA (Information technology Promotion Agency)
Sample: 258 Major Software Houses in Japan and Member of JISA (Japan Information
technology Service industry Association)

Embedded Software Crisis

Safety and Reliability Become Job #1

➡ From *Good-Enough Quality* to *eXtremely High Quality*

➡ **Safety-Critical Embedded Software**

➡ **Mission-Critical Software Systems:**
 $10^{}(-6)$**

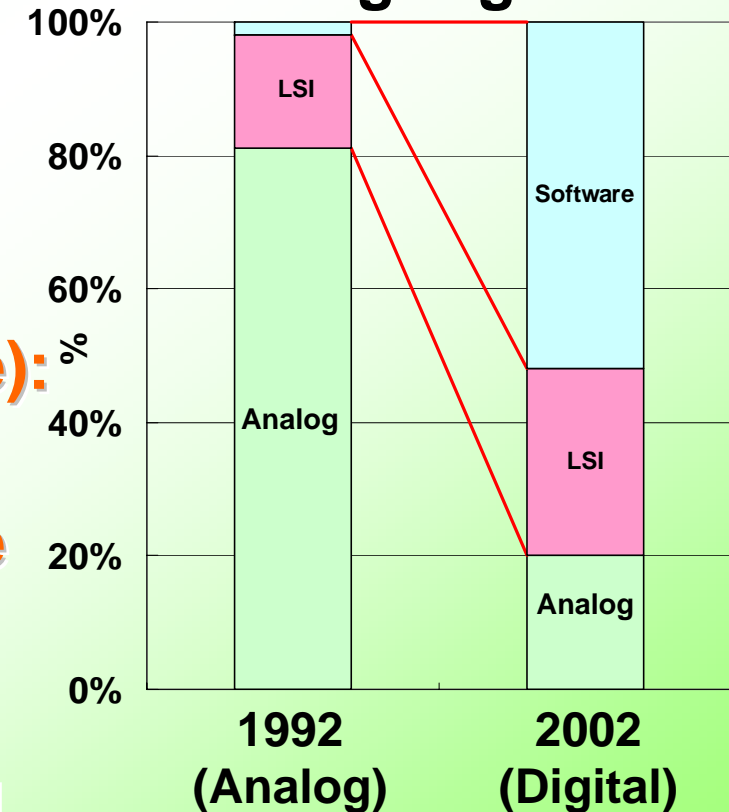
➡ **Safety-Critical Embedded Software**

👉 **Commercial Airplane (Fly-By-Wire):**
Flight Control Software= $10^{}(-10)$**

👉 **Automobile (Drive-By-Wire): Drive**
Control Software= $10^{}(-12?)$**
[Impact of Mass Production]

➡ **Software is Dominant in Embedded**
Systems: Consumer Electronics

Development Cost
Distribution
of Analog/Digital TV



Source: Presentation by Y. Imai
at IPA Symposium, Oct. 2003.

Embedded Software Crisis

Social Responsibility

👉 Japanese Software and Service Industry

👉 Revenue: ~\$140B, Employee: 560,000

👉 Embedded Software Industry

👉 Revenue: ~\$2B(\$35B for Systems), Employee: 150,000

👉 Embedded Software Development

👉 Dominated by Small-Medium Size Companies

👉 Few People with Software Engineering Education

👉 Industry Structure

👉 Hierarchical Industry Structure

👉 **Low Competitiveness**

👉 **No Technical Incentive**

👉 Offshore Development

Offshore

Small-Medium Software Companies

Plan & Mgmt

Big(10~20)

Design

Subsidiary Independent
of Big Soft. Houses

Implement







Source: MEIT Survey

Challenges and Our Strategies We Must Realize and Respond

View as Structural Problem

-  Not Coincidence, Not Specific Companies, (Not Country), ...
-  But Whole Software Industry and Research Community

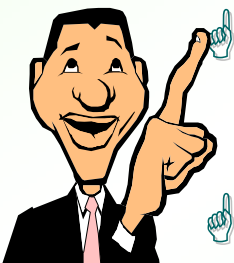
Strategic Approach

-  Long-Term Strategic Goal
 -  **No Short-Term “Naive Competitive” Funding**
-  Address Essential Issues in Software Engineering
 -  **Requirements, Reuse, Metrics, Visualization**
 -  **Scientific and Quantitative Approach**
 -  **Business Practice (Might be Specific to Japan)**

User/Customer Involvement

-  **Nothing Change without Customer’s Demand**

Lessons Learn from Past Failures of Government Projects



Challenges and Our Strategies

SEC (Software Engineering Center)

Sponsored by MEIT: Ministry of Economics, Industry and Trade

 Planning since March 2003 (Many Proposals before That)

 To be Officially Open on Oct 1, 2004,

 1st Year Budget: Some \$12M

Dedicated to Software Engineering Research & Practice

 Focus on Practice, and Collaboration between Industry and Academia

Major Area of Work

 Domains: Enterprise and Embedded Software

 Technologies

 Requirements Engineering

 Architecture & Components/Services

 Agile Development and Process Improvement

 Empirical Data Collection and Analysis

 Project/Quality Management

 Software Engineering Education

Challenges and Our Strategies

Global Collaboration

Concerning and Similar Activities Arising at Different Countries

- UK: RAEng Report with Possible Plan of SEI and London Software System (Imperial College, etc)
- US: PITAC Report [1999] and NIST Report on Inadequate Testing [2002]
- Known Success Stories: SEI(USA), Fraunhofer IESE (Germany)

Need Global Collaboration

- Ex: Impact Project (Osterweil, et al.)

Need to Convince Entire Society

- Importance of Software Engineering





Thank You