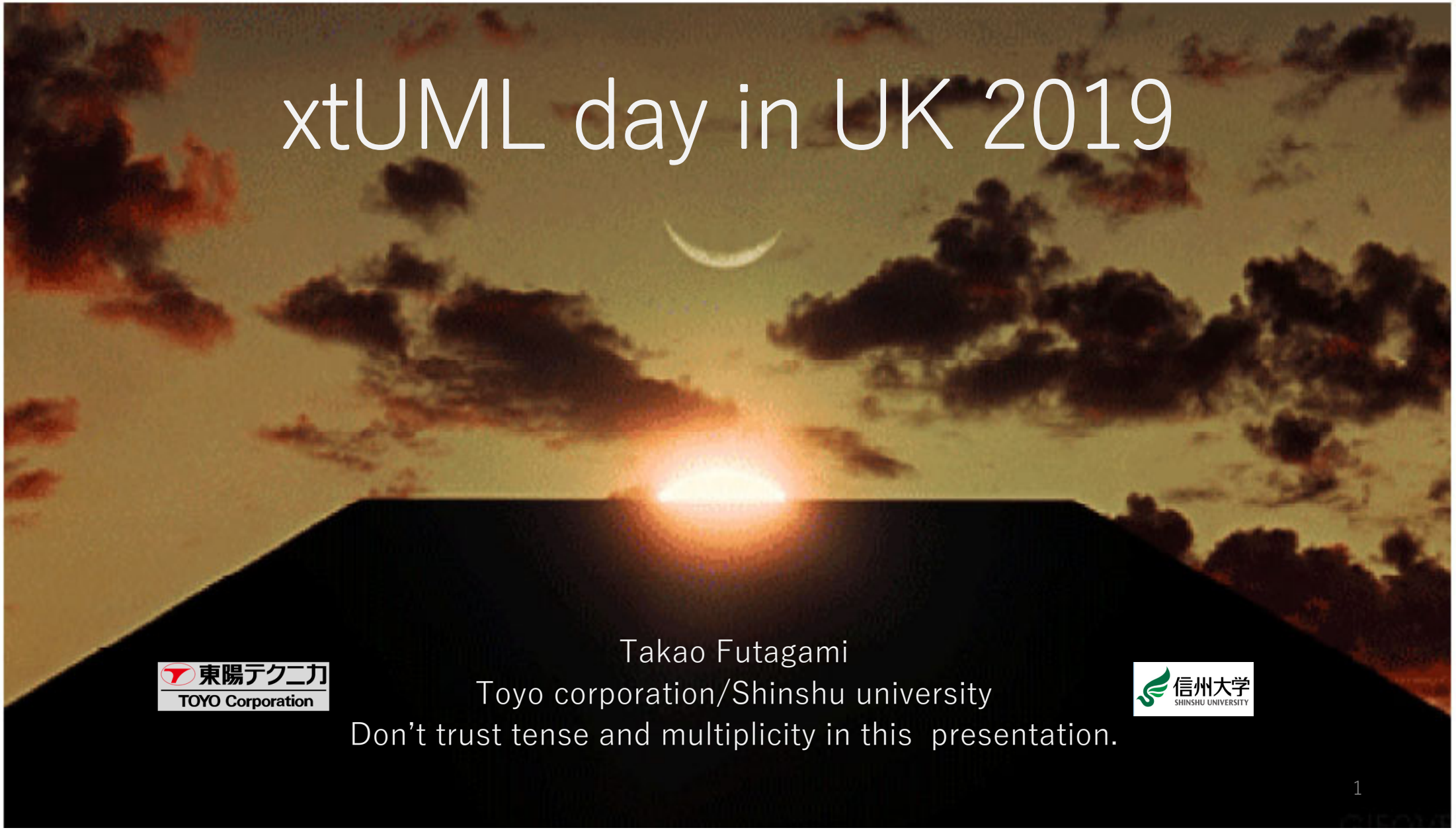
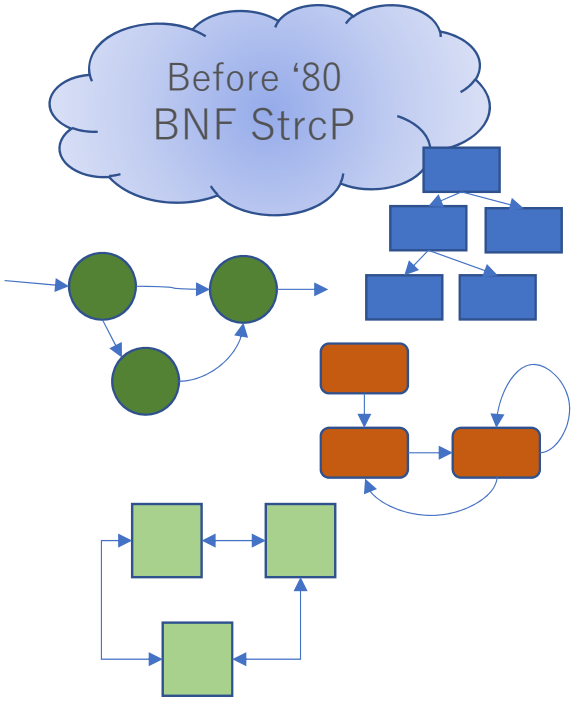


xtUML day in UK 2019

Takao Futagami
Toyo corporation/Shinshu university
Don’t trust tense and multiplicity in this presentation.

Decades of software engineering



- ~ ‘80s lexical and structural programing
- ‘80s SA/SD Structured method Morphing to OO
- ‘90s >33 OO Kingdoms OO medieval times
- ‘00s Unified/Process focused. Tool Imperialism / Dark age

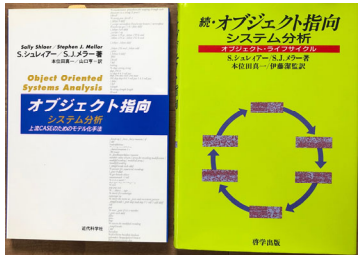
Building Cathedral
<MLOC world

Capability/Maturity
Audit/Inspection/Test

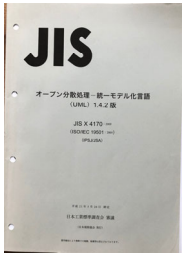
- ‘10s Agility focused Renaissance

Agile
Succeeds if member is skillful

SM and Japanese embedded software archeology







ET-Robocon

ESS-robot-challenge

‘80 ‘90 ‘00 ‘10 ‘20

Dawn of ET | CASE tool | Unification | Dark age | Renaissance



アウストラロピテクス
ジンジャントロプス





American
Imperialism

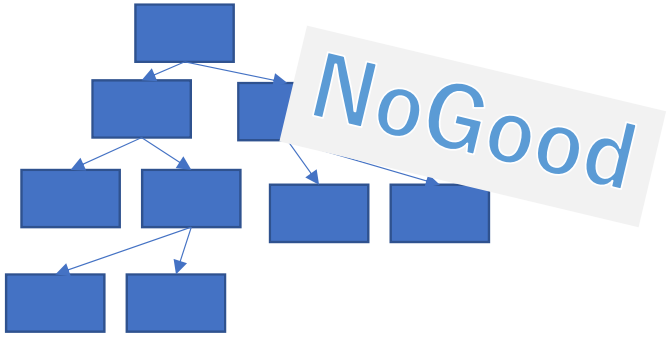


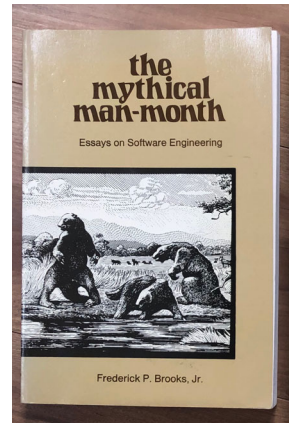
Late ‘70s in Japanese lab and office

- Computer was machine for computing and not for automation.

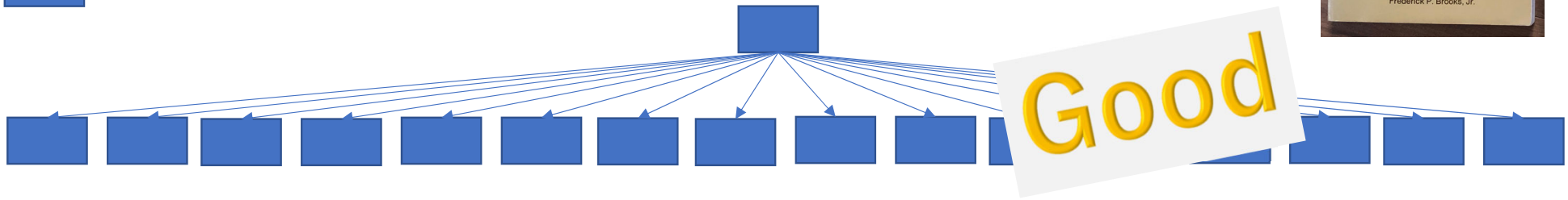
Just translate customer’s spec and equation into program.
Program doesn’t need design at all.

$$\frac{d\sigma}{d\Omega} = \frac{r_e^2}{2}\left[\frac{1}{1+\alpha(1-\cos\theta)}\right]^2\left[1+\cos^2\theta+\frac{\alpha^2(1-\cos\theta)^2}{1+\alpha(1-\cos\theta)}\right]$$





Place all subroutines in only one layer
to improve reusability



‘80s-’9x a few success and lots of disasters

- 70% of project was not completed actually
- Software Engineering arrived with CASE tools
- The word **火消 “Hikeshi”** was used for software fireman



構造化分析
セミナーご案内

講師
Dr.CliveBoughton

東陽テクニカ
TOYO Corporation

‘95 OO enlightenment by Sally and Steve

PROJECT TECHNOLOGY
Shlaer-Mellor Method

Dear Takao,

Here is the paper I mentioned.

(Note; I switched the terms "Behavioral" and "Structural" in my talks).

Thank you for your hospitality

Steve

Object-Orien
Design in Ada
Transformation
Strategy Based o
Object-Oriented
nalysis
ill



Documentation Guidelines
for Ada Object-Oriented Development
Under 2167A

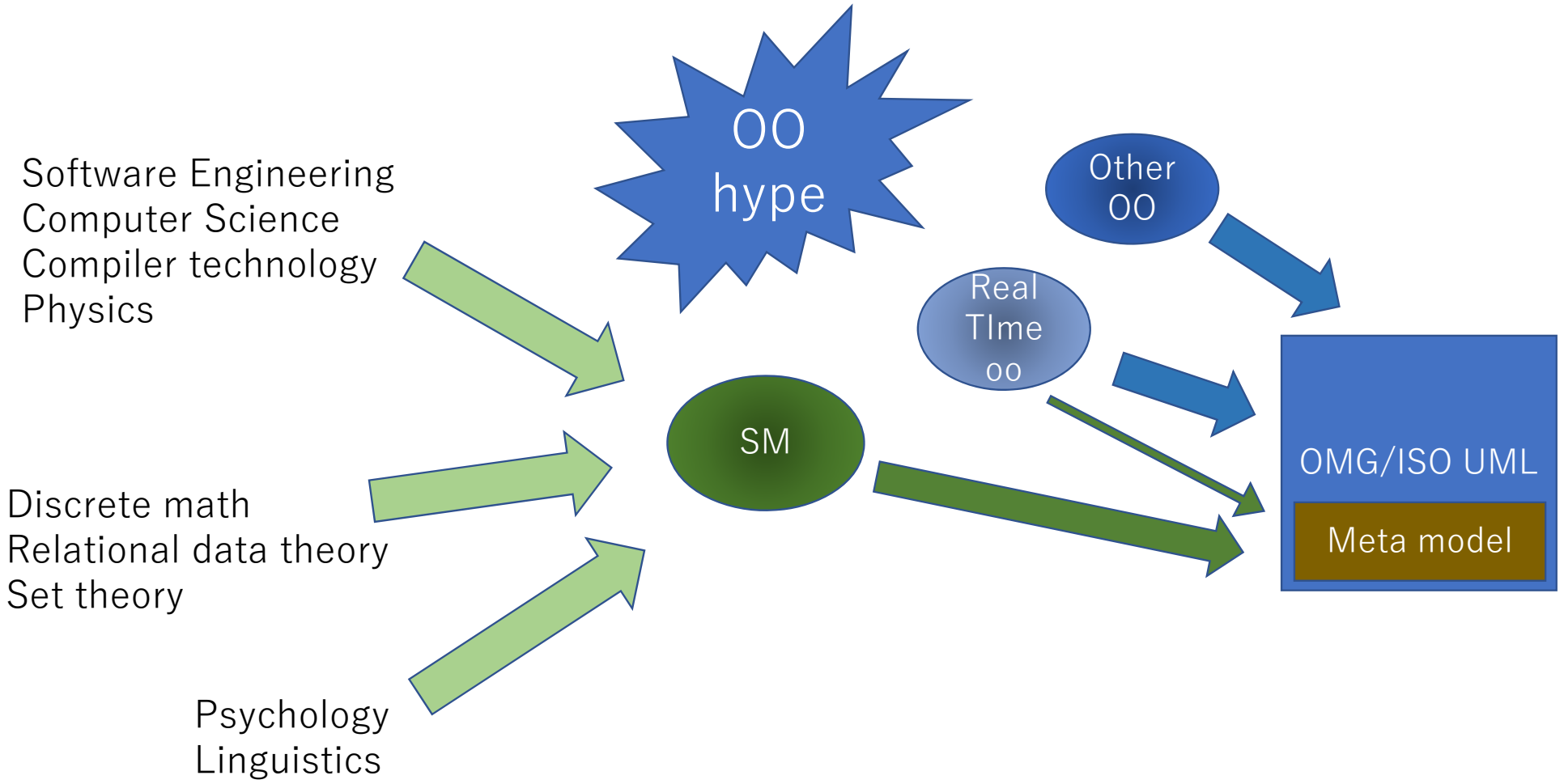
Prepared by:

Fredrick N. Hill
Stephen J. Mellor
Klaasy de Nevers
Sally Shlaer

Project Technology, Inc.
2560 Ninth Street, Suite 214
Berkeley, California 94710

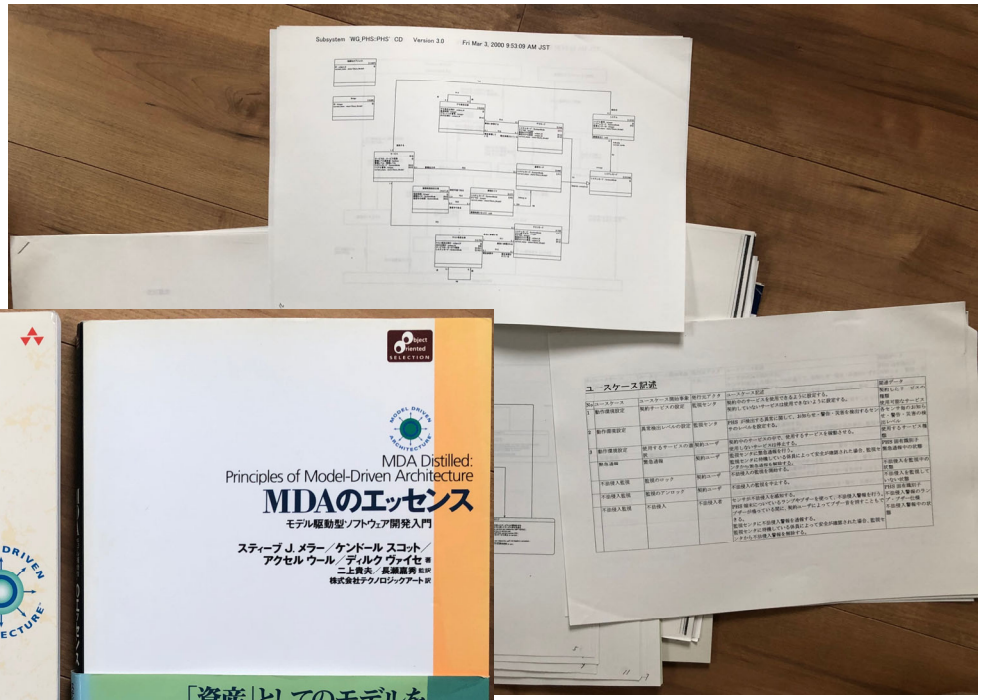
August 24, 1989

Evolution to SM then SM to xtUML



Effort to let people be understood SM.







Big bang of application development





- Train management: ‘95-’00
- Medical device: ’97
- Electric power generator: ‘98-’00
- 10^7 LOC challenge ‘95-



PRC Train management system: ‘95-’ 99

- Strictly safety-oriented product
- 500 classes on BP generates C++ code





Medical device: ‘97

- Intravenous infusion controller with very small footprint

O_d_as.c - メモ帳

ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
/*******************************************************************
 * File: O_d_as.c
 *
 * Asynchronous services for the following domain:
 * Domain:     O
 * Repository: NEC_ooa_timer.ooa
 *
 * Notice:
 *   (C) Copyright 1999 ROX Software, Inc.
 *   All rights reserved.
 *
 * Model Compiler: MC3020  V1.1.0
 * Serial Number:  302011099070502
 *
 * Warnings:
 *   !!! THIS IS AN AUTO-GENERATED FILE. PLEASE DO NOT
 *******************************************************************
```

- Not cancer but headache
  分析からコードへ
  - If compiler rejects “Internationalization”
    you “ I18s”
    me “国際語化”

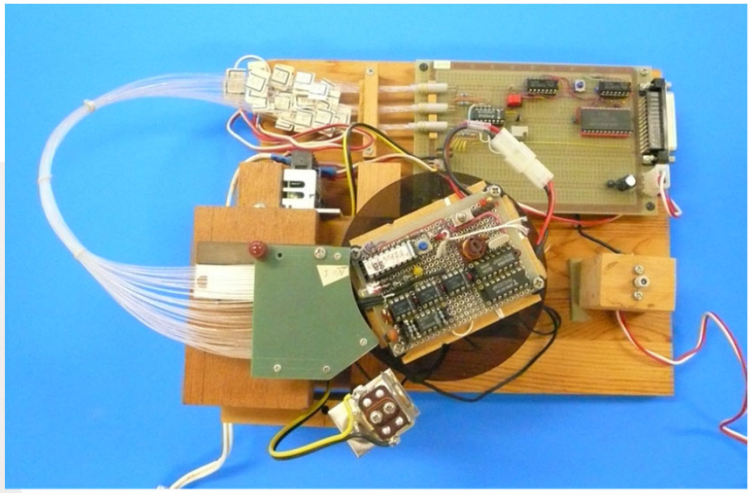
イル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

```
 /* FOR EACH timer IN timer_set */
 {
   Escher_Iterator_s iter6_OTIM;
   Escher_IteratorReset( &iter6_OTIM, &v3 );
   while ( (v4 = (O_OTIM_s *)Escher_Iterator
   {
   /* IF ((timer.使用中) AND (timer.発火予定
   if ( ((v4->m_is_using) && (v4->m_firing_c
   {
     /* ASSIGN 発火時送信イベント = timer.発
     v5 = v4->m_sending_event;

     /* GENERATE 発火時送信イベント */
     Escher_SendEvent( (OoaEvent_t *)v5 );
```

Electric power generator: ‘98-’00

- SM is used for
  - Software analysis model
  - Patent idea generation



Application of xtUML to Multi-Function Printer

xtUML2019 Days UK

September 12th 2019

Fuji Xerox Co., Ltd.

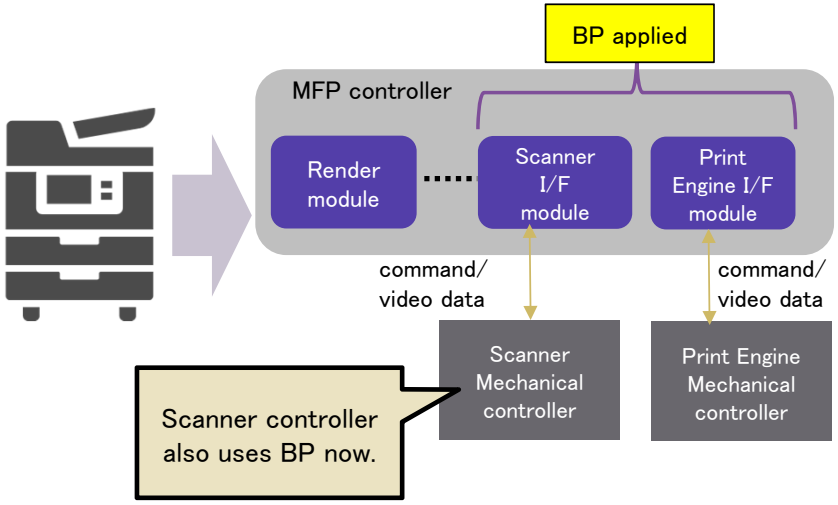
Software & Electronics Development Group

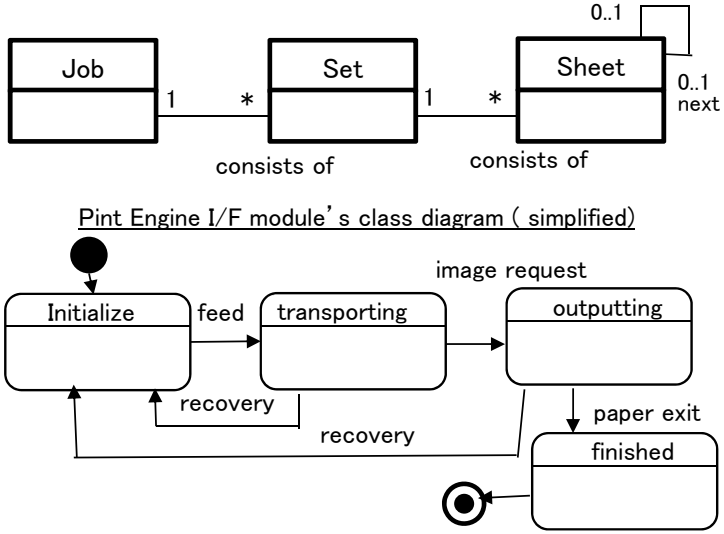
Manager

Yuki Tsuchitoi

Background of application of xtUML to MFP

- Multi-Function Printer(MFP) / Single Function Printer(SFP) must handle multiple papers in printing, scanning and many states. Besides that, must handle unexpected event such as paper jam and cover open, stop/cancel instruction and so on.
- This kind of multi instance and event handling cannot be implemented easily with normal programming language like C.
- As BridgePont/xtUML can easily handle these multiple instances and the states, it is suitable for developing these modules.
- We began to use BridgePoint for developing Print Engine I/F module and Scanner I/F module in1999. Our first SFP product using BridgePoint was launched in 2000, MFP was in 2001





State Machine of 'Sheet' (simplified)

Overview of our development using BP

- Migration History
  - In 2001–2004 : Used Old BridgePoint with MC2012 (C++ based model compiler)
  - In 2004 : Changed model compiler from MC2012 to MC3020 (C based model compiler) Without changing models.
  - In 2015–2016 : Changed over to Eclipse based BridgePoint 4.2 supported by OneFact.

- Metrics

number of domains	number of modelers	number of classes per domain	number of line of actions (w/o comment) per domain	number of line of generated codes per domain	ratio of generated code / action per domain	rate of code from model per domain
9	> 10	8 to 119	1.2K to 2.5K	6K to 90K	300% to 450%	23% to 68%

- Current State
  - Members using BP do not review generated code, but only action code.
  - They have confidence that there are no differences between model and code.
  - Small number of member (BP Architect) can maintain the BP's model compiler for all domains. Actually, only one person maintains model compiler currently.
  - It is very convenient that common changes (like debugging, log) among all domains can be controlled by changing only a model compiler.
  - Made internal modeling education materials (mainly class diagram)based on xtUML.
- Issues and Further Activities
  - Making full use of Verifier : modelers who can use Verifier effectively are limited.
  - Tool issues: BP4.2 have performance issues in interpreter based model compiler and Verifier.
    Need evaluation of latest BP and plan of migration.

21st century problem in Japan
Obvious trench between elite and ordinary people

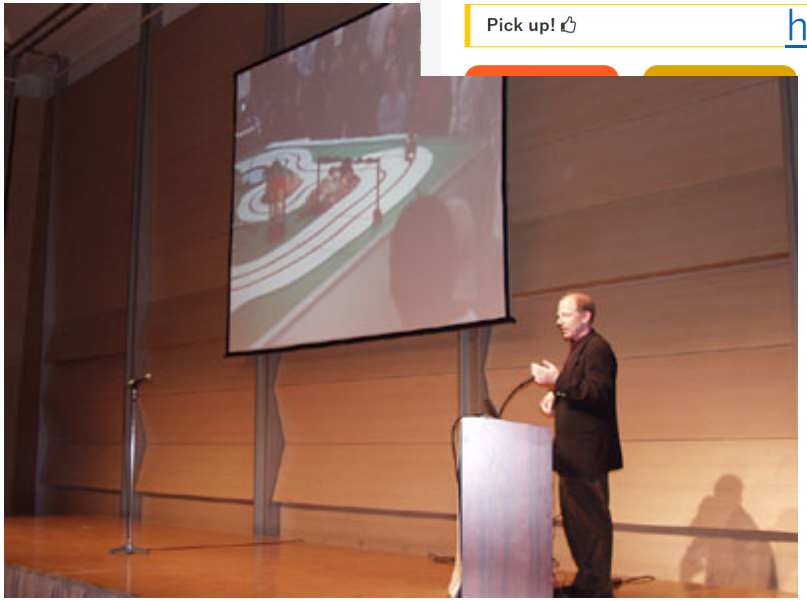
- The more champion team succeeded the less others followed.
- Only the first team could enjoy long term learning opportunity.
- We needed some sort of mass education for industry people.
  - UML robot contest
  - MDD robot challenge

2001 UML robot contest for embedded software

- OMG and Japanese companies co-work to create modeling contest.
- Base education material came from Lego path finder at PT exhibition booth.
- Leon was the first judge member







2004 MDD robot challenge for system development

Lego bundled modeling is not enough





International education opportunity

- Thanks to SMUG in 2004,2005





2004 Government noticed about embedded software industry.



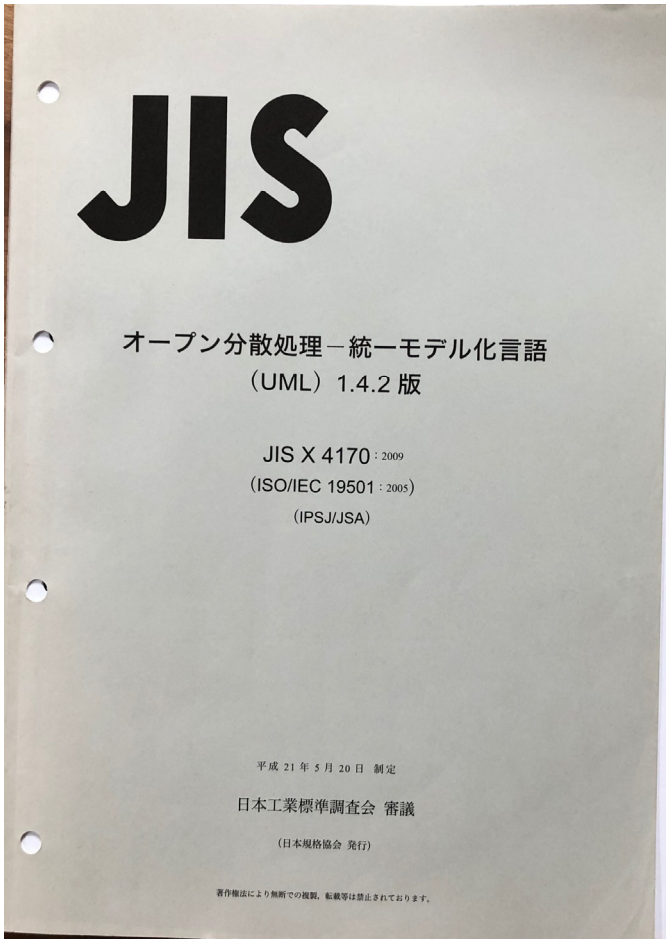
文部科学省 MINISTRY OF EDUCATION, CULTURE, SPORTS, SCIENCE AND TECHNOLOGY-JAPAN

Keidanren
Japan Business Federation

METI

- METI initiated committee for embedded software industry analysis.
- Keidanren expressed necessity of embedded engineer.
- Finally,
  - UML robot contest is inherited by METI extension as ET-robocon
  - MDD robot challenge is inherited by MEXT as ESS robot challenge

2009 UML ⇒ Japanese industry standard



JIS

オープン分散処理－統一モデル化言語
（UML）1.4.2 版

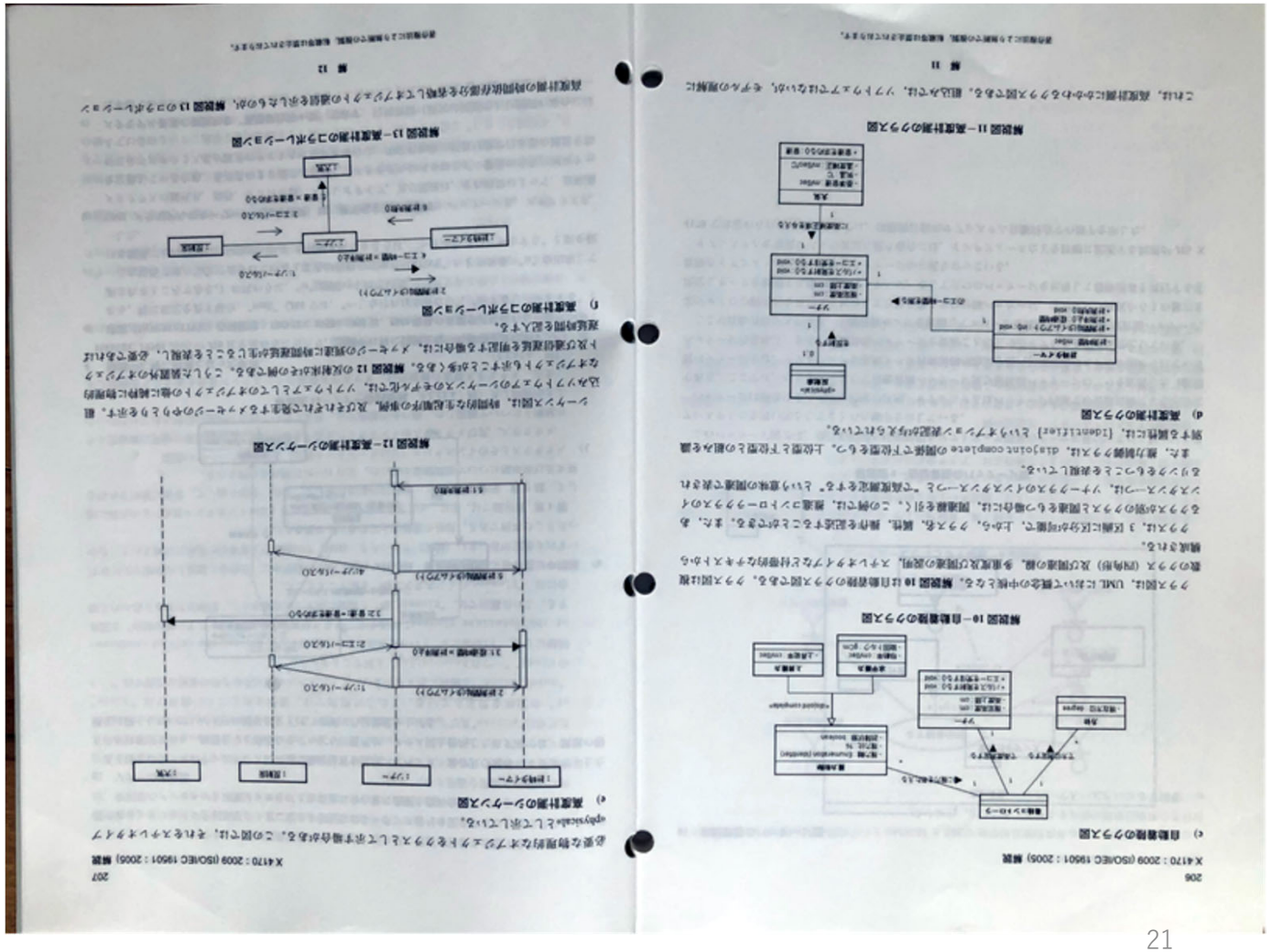
JIS X 4170：2009
（ISO/IEC 19501：2005）
（IPSJ/JSA）

平成 21 年 5 月 20 日 制定

日本工業標準調査会 審議

（日本規格協会 発行）

著作権法により無断での複製、転載等は禁止されております。



2005-2015 Imperialism / Dark age

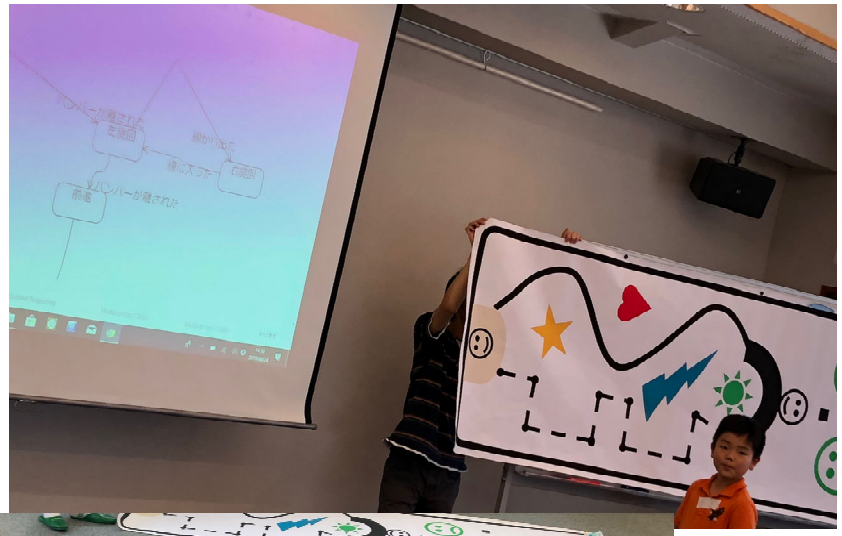
- Software Engineering tools became food for M&A business.
- End of slow but steady improvement.

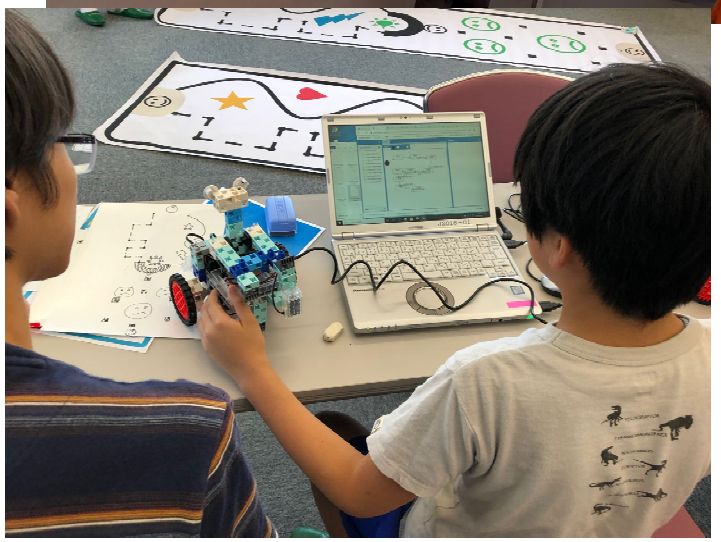


Renaissance

- New age of investment for new generation.







東陽テクニカ
TOYO Corporation

信州大学
SHINSHU UNIVERSITY