

## Note

This document is an English translation based on [the Japanese version](#). If there are any discrepancies, inconsistencies, or contradictions between the translation provided and the Japanese version, the Japanese version shall take precedence.

# Global Trend Research Report on Software Engineering

December 10, 2024  
Software Engineering Group  
Digital Engineering Department  
Digital Infrastructure Center

## ◆ Background

As the phrase "Software Is Eating the World" suggests, software has become a critical source of competitiveness for countries and companies. It has evolved into a serious issue that impacts the competitiveness of entire industries, not just the IT sector.

Software engineering has reached a major turning point, prompting the IPA to fundamentally review its past software development research and examine cutting-edge software policies in light of the latest trends in the United States and Europe.

The results of the "FY2023 Questionnaire Survey on Software Development" released by IPA in January 2024 revealed that software development in Japan remains entrenched in outdated methodologies or thinking such as man-months and SLOCs, and that document-based development is still deeply rooted in Japan, while the U.S. and Europe are working on model-based development methods.

## ◆ Objectives

Accepting the results of the "FY2023 Questionnaire Survey on Software Development", this research was conducted to identify as accurately as possible the differences in the trends of software engineering between in Japan and in the United States and Europe.

The findings will serve as fundamental information on the current state of advanced software engineering and will help determine the direction of software engineering policies in Japan.

## ◆ TARGETS :

Advanced software companies, institutes, and organizations, mainly in the U.S. and Europe

## ◆ CONTRACTOR :

Scaleout Inc. (<https://scaleout.tv>)

## ◆ PERIOD :

June - September 2024 (survey conducted by contractor)

September - October 2024 (IPA to organize results and conduct additional research)

## ◆ METHODS :

Literature and web reviews, online interviews (15 participants)

- [CMU-SEI](#) affiliates
- [OMG](#) affiliates
- [IFPUG](#) affiliate
- [ISBSG](#) affiliate
- [NESMA](#) affiliate
- [COSMIC](#) affiliate
- [IEEE](#) affiliate
- Advanced software company affiliates

Note: In principle, interviews were conducted on condition of anonymity.

# Current Situation of Software Engineering in Japan and Hypotheses in the U.S. and Europe

- ◆ Based on the results of the analysis of the "FY2023 Questionnaire Survey on Software Development", we have developed hypotheses in the U.S. and Europe about the unique situation in Japan.

## SITUATION(1) in Japan

Cost estimating is still based on experience and intuition.

### HYPOTHESIS (1) in the U.S. and Europe

To strategically drive the software business, new estimation methods are being considered, such as exploring methods based on historical project data.

## SITUATION(2) in Japan

Not a few want to move away from the man-month business model, under the guise of improving the situation of engineers.

### HYPOTHESIS (2) in the U.S. and Europe

The business model is different from that of Japan. Value-based business is more advanced, where some method (know-how) of software valuation is believed to exist.

## SITUATION(3) in Japan

Model-based requirement specification is not preferred, and document-based requirements definition is preferred.

### HYPOTHESIS (3) in the U.S. and Europe

Model-based development approach has spread from safety-critical industrial applications to the manufacturing sector and is now of growing interest to the enterprise sector.

## SITUATION(4) in Japan

There is a development tendency to "create" software that is tailored to the business rather than "use" existing components.

### HYPOTHESIS (4) in the U.S. and Europe

Since Fit to Standard is well-established, a culture is fostered in which operations are aligned with standards and components to be created are strategically determined.

- ◆ The following research themes were selected based on the current situation in Japan and hypotheses in the United States and Europe.

## 1. Software Development Planning Methods

- (1) Estimation Methodology
- (2) Projects Metrics Data Available for Estimation
- (3) How to Determine the Development Work Period
- (4) Contents and Purpose of the Development Project Plan

## 2. Software Valuation Methods

- (1) Pricing of SaaS (Software as a Service)
- (2) Case Studies of Value-based Compensation for Outsourcing

## 3. Modeling

- (1) Adoption Status of Modeling Approach
- (2) Modeling Languages
- (3) Interoperability between Models

## 4. Building Blocks (Microservices, Software Components, etc.)

- (1) Monolith to Microservices Migration Case Studies
- (2) API Conversion Case Studies of Legacy Systems

## Hypotheses in the U.S. and Europe

### 1. Software Development Planning Methods

To strategically drive the software business, new estimation methods are being considered, such as exploring methods based on historical project data.

### 2. Software Valuation Methods

The business model is different from that of Japan. Value-based business is more advanced, where some method (know-how) of software valuation is believed to exist.

### 3. Modeling

Model-based development approach has spread from safety-critical industrial applications to the manufacturing sector and is now of growing interest to the enterprise sector.

### 4. Building Blocks

Since Fit to Standard is well-established, a culture is fostered in which operations are aligned with standards and components to be created are strategically determined.

## Research Findings

- While there is no difference between Europe and the U.S. and Japan in the technical aspects of estimation methods, the U.S. and Europe are more advanced in the strategic aspects.
  - There are studies and efforts to develop new methods of estimating costs by analogy with historical project data similar to the subject project.
  - Despite the expectations for estimation methods for agile development and component-based development, notable examples do not seem to have emerged yet.
- As an example of value-based business, we researched B2B SaaS business. While a usage-based pricing model has generally been used, recent blogs are featuring the outcome-based pricing models that linked to the value of the customer's business.
  - We were unable to find any examples of value-based compensation in outsourcing.
- Model-based development approach has been established in safety-critical areas such as the automotive and aerospace industries and now appears to be spreading to manufacturing areas, but it has not yet been widely adopted in the enterprise sector.
  - In the study of enterprise sector, the OMG's Unified Architecture Framework (UAF) is gaining attention and appears to be targeting interoperability at the organizational and business levels.
- Software that creates new value is developed strategically. If it is found that a combination of existing software creates new value, the existing software is used while it is broken down into components.
  - If the strategically developed software is a monolith, it is broken into components to create even more value.

# 1. Software Development Planning Methods

- (1) Estimation Methodology
- (2) Projects Metrics Data Available for Estimation
- (3) How to Determine the Development Work Period
- (4) Contents and Purpose of the Development Project Plan

## Summarizing the Research Findings

- While there is no difference between Europe and the U.S. and Japan in the technical aspects of estimation methods, the U.S. and Europe are more advanced in the strategic aspects.
- There are studies and efforts to develop new methods of estimating costs by analogy with historical project data similar to the subject project.
- Despite the expectations for estimation methods for agile development and component-based development, notable examples do not seem to have emerged yet.

# Theme 1: Software Development Planning Methods

## (1) Estimation Methodology

### ■ Exploring New Methodologies Through Analogies from Historical Project Data

#### (1) Study of a New Estimation Method: Capability-Based Software Cost Estimation (CaBSCE)

- CMU-SEI started a trial of a new estimation method "Capability-Based Software Cost Estimation (CaBSCE)" for DoD.
- To estimate the cost and schedule of a software development project, project data with similar functions and characteristics is extracted from the repository of software development project data and analyzed.

References: [CMU-SEI, "Capability-Based Software Cost Estimation \(CaBSCE\)", September 4, 2024](#)

[Anandi Hira, "Capability-Based Software Cost Estimation \(CaBSCE\): Proposing a New Method to Estimate Software Costs", April 30, 2024](#)

#### (2) Case Study on Consulting Services for Estimating Costs Using Historical Project Data

- LedaMC, a Spanish consulting firm, is developing an estimation consultation business based on its software development project data.

Reference: [LedaMC, "SaaS APP for Software Project Estimation"](#)

### ■ The U.S. and Europe are More Advanced in the Strategic Aspects

- The International Cost Estimating and Analysis Association (ICEAA) in the United States has initiated the Software Cost Estimating Certification (SCEC) program.
- Since strategies for national defense and business management in the U.S. are centered on software, estimation methods have been systematized on the management side and training of estimation experts has begun.

Reference: [International Cost Estimating and Analysis Association, "The Certification Program"](#)



# Theme 1: Software Development Planning Methods

## (1) Estimation Methodology (continued)

### ■ Agile Development Metrics Not Yet Standardized

- Agile metrics are growing in popularity, but the usage guide has not yet been standardized. Agile developers focus on self-improvement rather than benchmarking against other projects.

Interviewees: Affiliates from an organization or research institute that studies software cost estimation

### ■ Study of Component-based Estimation Methods is Not Active

- No recent case studies or research on component-based software cost estimation were found.

# Theme 1: Software Development Planning Methods

## (2) Project Metrics Data Available for Estimation

### ■ The Only ISBSG to Collect and Publish Metrics Data Globally

- The International Software Benchmarking Standards Group (ISBSG), an Australian non-profit organization, was established in 1997 and publishes data provided by cost estimation consulting firms and others.
- Number of cases: approx. 10,000 (\*Reference: IPA's own collected and accumulated data is approx. 5,000)
- While much data has been gathered on waterfall development projects, not enough data has been gathered on agile development (The same is true for the IPA).

Reference: [ISBSG, "A global & independent source of software metrics data for IT projects"](#)

\*Only ISBSG publishes data collected worldwide.

\*CMU-SEI appears to collect metrics data from government organizations, but it is not publicly available.

\*Some estimating consultancies collect their own data, but this is not made public.

# Theme 1: Software Development Planning Methods

## (3) How to Determine the Development Work Period

### ■ What Can Be Delivered in a Given Period Rather Than Accumulating Efforts

- The approach to software development planning for providing value to customers in an agile manner is fundamentally different from the past. The project schedule is not determined by the accumulated man-hours, but rather by what can be delivered in a period.
- The purpose of software development has shifted from achieving QCD targets of the project to delivering value to the customer.
- The trend in deciding the schedule and delivery timeframe for a software development project is changing from deciding in accordance with standard rules to deciding flexibly on a project-by-project basis.

Interviewee: A senior software engineer developing a SaaS platform for a major global technology company

# Theme 1: Software Development Planning Methods

## (4) Contents and Purpose of the Development Project Plan

### ■ Capturing Requirements in Customer-Value-Oriented Software Development

Unlike traditional waterfall development, agile development, which aims to provide value to customers, produces documents such as BRD, MVP, and even MLP.

#### (1) BRD (Business Requirement Document)

- The following articles introduce the Business Requirement Document (BRD), which focuses on the user experience when describing requirements. BRD differs from traditional requirements documents, which are written from a technical perspective.

References: [asana, "Business requirements document template: 7 key components, with examples", January 21st, 2024](#)  
[Indeed Editorial Team, "Business Requirements Document: Definition and Tips", August 18, 2024](#)

#### (2) MVP (Minimum Viable Product) and MLP (Minimum Lovable Product)

- In hypothesis-testing development, the minimum necessary functionality (MVP) is implemented and released, but Amazon further defines and implements the minimum functionality (MLP) that customers love.
- It is said that hypothesis-driven development implements and releases the minimum viable product (MVP). At Amazon, they further define and implement the minimum lovable product (MLP).

References: [Doron Katz, "Minimal Lovable Product \(MLP\) #amazon #experience #customerObsession", Jun 17, 2021](#)  
[Komal Singh, "After MVP", Nov 7, 2022](#)

## 2. Software Valuation Methods

- (1) Pricing of SaaS (Software as a Service)
- (2) Case Studies of Value-based Compensation for Outsourcing

### Summarizing the Research Findings

- As an example of value-based business, we researched B2B SaaS business. While a usage-based pricing model has generally been used, recent blogs are featuring the outcome-based pricing models that linked to the value of the customer's business.
- We were unable to find any examples of value-based compensation in outsourcing.

# Theme 2: Software Valuation Methods

## (1) Pricing of SaaS (Software as a Service)

### ■ Interests in Outcome-Based Pricing for B2B SaaS

- B2B SaaS companies will be expected to contribute to their clients' performance as companies become more focused on ROI (return on investment).

References: [Dominic Perrett, Sam Shinner, Jade Kahn, Todd Anstett, "The Rise of Outcome-Based Pricing in SaaS: Aligning Value With Cost ", September 13, 2024](#)

: [Cases 1, 2].

[James Wood, "The spectrum between outcome-based and usage-based pricing metrics: Insights from Intercom's Fin AI and more ", April 24, 2024](#) : [Case 2].

[Bright Reach, "Salesforce Agentforce and the Shift to Outcome-Based Pricing", September 29, 2024](#) : [Cases 2,3,4].

#### Case Study 1 : [Riskified](#)

- Service provider that detects fraudulent transactions during e-commerce payments, charging only when a fair transaction is completed and not charging when fraud is detected. A similar company is Lithuania's [iDenfy](#).

#### Case Study 2 : [Intercom](#)

- Messaging platform provider that mediates B2C business. With its AI chatbot service, only charges when the query is resolved.

#### Case Study 3 : [EvenUp](#)

- Provides a service that uses AI to generate legal documents for personal injury lawyers, with a fee based on the number of documents and pages.

#### Case Study 4 : [Salesforce Agentforce](#)

- AI platform provided by Salesforce, which can automate the design of business processes such as sales, customer support, and marketing. Its pricing model has shifted from the traditional seat-based model to one that charges per query to the AI agent.

# Theme 2: Software Valuation Methods

## (2) Case Studies of Value-based Compensation for Outsourcing

### ■ No Case Studies Found

It is said that self-employed or freelance workers with high software engineering skills are paid according to their skills. We tried to find such cases through interviews and other methods, but were unable to find any within the limited timeframe of our research.

# 3. Modeling

- (1) Adoption Status of Modeling Approach
- (2) Modeling Languages
- (3) Interoperability between Models

## Summarizing the Research Findings

- Model-based development approach has been established in safety-critical areas such as the automotive and aerospace industries and now appears to be spreading to manufacturing areas, but it has not yet been widely adopted in the enterprise sector.
- In the study of enterprise sector, the OMG's Unified Architecture Framework (UAF) is gaining attention and appears to be targeting interoperability at the organizational and business levels.



# Theme 3: Modeling

## (1) Adoption Status of Modeling Approach

### ■ Expanding to digital twins and ecosystems from safety-critical areas

- In December 2023, the DoD published [DoDI 5000.97 "Digital Engineering"](#), which basically directs the use of MBSE and modeling languages, and directs that the means of communication for system development be shifted from "documents" to "models". The DoD has instructed that the means of communication for system development should be shifted from "documents" to "models".
- Models not only represent the system, but are also essential for reasoning, analysis, and simulation.
- The model includes a digital twin concept that synchronizes with models of existing systems, such as nuclear power plants and aircraft, for predictive maintenance and anomaly detection.

Interviewee: CMU-SEI affiliate

### ■ Unknown in Areas Requiring Agility and Adaptability

- It seems that development, which flexibly accepts frequent additional or changing requirements, does not use processes such as MBSE because it is not lightweight.

Interviewee: A senior software engineer developing a SaaS platform for a major global technology company

### ■ The OMG's SysML v2 Hopes to be a Game Changer

#### (1) SysML v2 Overview

- It is based on a proprietary meta modeling language (Kernel Modeling Language). SysML v1 was an extension of the UML base.
- In addition to graphical notation, models can be described in a formal language based on textual syntax.
- A common interface API is provided.
- The project is scheduled to open to the public by the end of 2024.

Reference: [Sanford Friedenthal, "SysML v2 Basics", January 28, 2024](#)

#### (2) Use of the Current Version of SysML v1

- According to the results of the ["FY2023 Questionnaire Survey on Software Development"](#) published by the IPA in January 2024, the use of SysML v1 in Japan is lower than that of UML.
- In Europe, Thales' Capella was more popular than SysML v1, which has restrictions, but Capella users are interested in connecting to SysML v2.

Interviewee: OMG affiliate

- The OMG website has published the INCOSE meeting materials that compare Thales's Capella and SysML.

References: [Torrance INCOSE IW Meeting - January 2017 SysML v2 RFP Working Group Meeting](#)

[Stéphane Bonnet, "Arcadia/Capella: looking back at our implementation issues CONSIDERATIONS FOR SYSML V2", January 28 2017](#)

### ■ OMG's UAF as a Comprehensive Business-Level Modeling Language

#### (1) UAF (Unified Architecture Framework) Overview

\*V1.2 @ July 2022 is the latest. First edition V1.0 @ October 2017 published.

- It consists of Part 1: Domain Metamodel (DMM) and Part 2: UAF Modeling Language (UAFML).
- The following Architecture Framework is integrated.
  - U.S. Department of Defense Architecture Framework (DoDAF)
  - UK Ministry of Defence Architecture Framework (MODAF)
  - NATO Architecture Framework (NAF)
- The requirements of the military domain are considered to cover 90% of the requirements of the business sector.
- UAFs are attracting attention because modeling of organizational and business structures is necessary to realize digital ecosystems.

Reference: [OMG, "Introduction to the 'Unified Architecture Framework®\(UAF®\)'"](#)

#### (2) Another Architecture Framework

- The Open Group published TOGAF version 1.0 as a framework for enterprise architecture in 1995.

Reference: [The Open Group, "Welcome to the Digital Edition of the TOGAF Standard"](#)

- The Open Group discussed the comparison of TOGAF and UAF and a mapping of TOGAF to UAF at the 2019 Architecture Forum.

Reference: [The Open Group, Denver 2019: Proceedings - Architecture Forum: The TOGAF® Standard to OMG UAF Mapping](#)

### ■ OMG's Completed BPMN Complemented by CMMN and DMN

#### (1) BPMN Business Process Model and Notation

\*No updates since V2.0.2 @ January 2014

Reference: [OMG, "Business Process Model & Notation™ \(BPMN™\)"](#)

#### (2) CMMN Case Management Model and Notation

\*No updates since V1.1 @ December 2016

Reference: [OMG, "Case Management Model and Notation™ \(CMMN™\)"](#)

#### (3) DMN Decision Model and Notation

\*V1.6 @ September 2024 is the latest. Continuously updated since first edition V1.0 @ September 2015.

Reference: [OMG, "Decision Model and Notation™ \(DMN™\)"](#)

#### (4) SDMN Shared Data Model and Notation

\*V1.0 beta 2 @ June 2024 is the latest. Being updated for official publication.

See [OMG, "About the Shared Data Model and Notation Specification Version 1.0 beta 2"](#)

- In the OMG Business Process Modeling Notations, the basic BPMN is already complete. However, BPMN is not suitable for cases where the process flow cannot be determined in advance (such as emergency management in the medical, fire, and police services), so CMMN can be used to supplement this.
- When using BPMN, CMMN, and DMN in the same area, common data is modeled using SDMN.
- This year, the OMG is focusing on the adoption of BPMN in healthcare.

Interviewee: OMG affiliate

### ■ Aiming for Interoperability between Different Layer Models

- The vision of the modeling research community and organizations is to enable a seamless transition from organizational (business) models to systems models to software models through a common modeling language.
- The OMG has published UML, SysML, and UAF as modeling languages for software, systems, and organizations, respectively, and SysML v2, the next version of SysML, will provide common APIs to promote interoperability.
- In addition, efforts are underway to link SysML to ontology definition languages such as Web Ontology Language (OWL) to enhance interoperability.

Interviewee: CMU-SEI affiliate

### ■ MBSE Approach Referenced Beyond Just Systems Modeling

- As the modeling scope expands from the systems level to include software and organizations, MBSE is referenced not only for systems modeling.
- CMU-SEI calls MBSE "Model-Based Software Engineering" when the emphasis is on software.

Interviewee: CMU-SEI affiliate

# 4. Building Blocks

## (Microservices, Software Components, etc.)

- (1) Monolith to Microservices Migration Case Studies
- (2) API Conversion Case Studies of Legacy Systems

### Summarizing the Research Findings

- Software that creates new value is developed strategically. If it is found that a combination of existing software creates new value, the existing software is used while it is broken down into components.
- If the strategically developed software is a monolith, it is broken into components to create even more value.

### ■ Strategic Migration from Newly Developed Monoliths to Microservices

The following e-commerce site or platform companies strategically adopted a monolithic architecture system with agility as a top priority to quickly gain market share. When market expansion became more evident, they moved to a microservices architecture to enable scalability, easily add and change functionality, quickly support security, and minimize disruption in the event of a failure.

#### Case Study 1: [Netflix](#)

- Provide streaming services over the Internet.

Reference: [Netflix, "Completing the Netflix Cloud Migration", 12 February 2016](#)

#### Case Study 2: [Uber](#)

- Operates taxi dispatch service, food delivery service, cargo transport service, etc. worldwide through a smartphone app.

Reference: [Adam Gluck, "Introducing Domain-Oriented Microservice Architecture", 23 July 2020](#)

#### Case Study 3: [Etsy](#)

- E-commerce mall for handmade and vintage products.

Reference: [Stefanie Schirmer, "API First Transformation at Etsy - Concurrency", 6 September 2016](#)

#### Case Study 4: [Amazon Prime Video](#)

- Streaming service provided by Amazon, Inc.

Reference: [AWS, Prime Video Boosts Scale and Resilience Using Amazon DynamoDB , 2018](#)

# Theme 4: Building Blocks

## (2) API Conversion Case Studies of Legacy Systems

### ■ Interoperating and Revitalizing Legacy Systems with APIs

The following referenced articles provide case studies of how legacy systems are increasing interoperability through an API gateway.

#### Case Study 1: [Finologee](#)

- Digital financial platform operator in Luxembourg
- In response to the European Union's Payment Services Directive (PSD2), this company developed a platform that allows banks to provide account information and payment instructions to third parties via API.
- Developed based on IBM API Connect®.

Reference: [IBM, "Fast-track digital compliance for the financial industry"](#)

#### Case Study 2: [TINE](#)

- Norway's largest producer, distributor and exporter of dairy products
- This company is integrating the legacy systems into the newly developed Integration Center in a long-term plan by building a new API gateway on the legacy systems, while continuing to operate the legacy systems, which are intricately intertwined.
- Developed based on IBM API Connect®.

Reference: [IBM, "Untangling A complex value chain"](#)



IPA