

# Hitachi AI Application Framework for Development and Operation of AI Applications

There is a rapidly growing need for the use of AI as a means of creating new value and innovation in social infrastructure and enterprise domain systems that directly connect to mission-critical businesses and handle vast amounts of data on a daily basis. Meanwhile, even when benefits are foreseen in the trial development phase, adding AI to the enterprise domain is a challenging obstacle with a high degree of difficulty as it involves requirements such as additional functions to enable actual operation adapted to daily changes. Hitachi has responded to this challenge by developing a new product called the Hitachi AI Application Framework. The product assists the development and operation of highly efficient and high-quality AI systems. This article looks at its aims and functional characteristics.

Tomonori Nakamura

Yoshitaka Terada

Chihiro Wada

Toshiyuki Ukai

## 1. Introduction

Since business-use artificial intelligence (AI) has finished its trial period and is now entering mainstream adoption, companies have started working on using AI to improve their existing business areas and create new ones. These efforts involve tasks such as automating and predicting operations using big data analysis, natural language processing, and image recognition.

However, AI is still largely only used for superficial business applications. Its use has not been fully realized in the enterprise domain—areas such as the foundational systems that make up the core of a company's business, or the mission-critical systems underpinning social infrastructure. One reason may be that enterprise systems have unique

characteristics different from the systems of other domains.

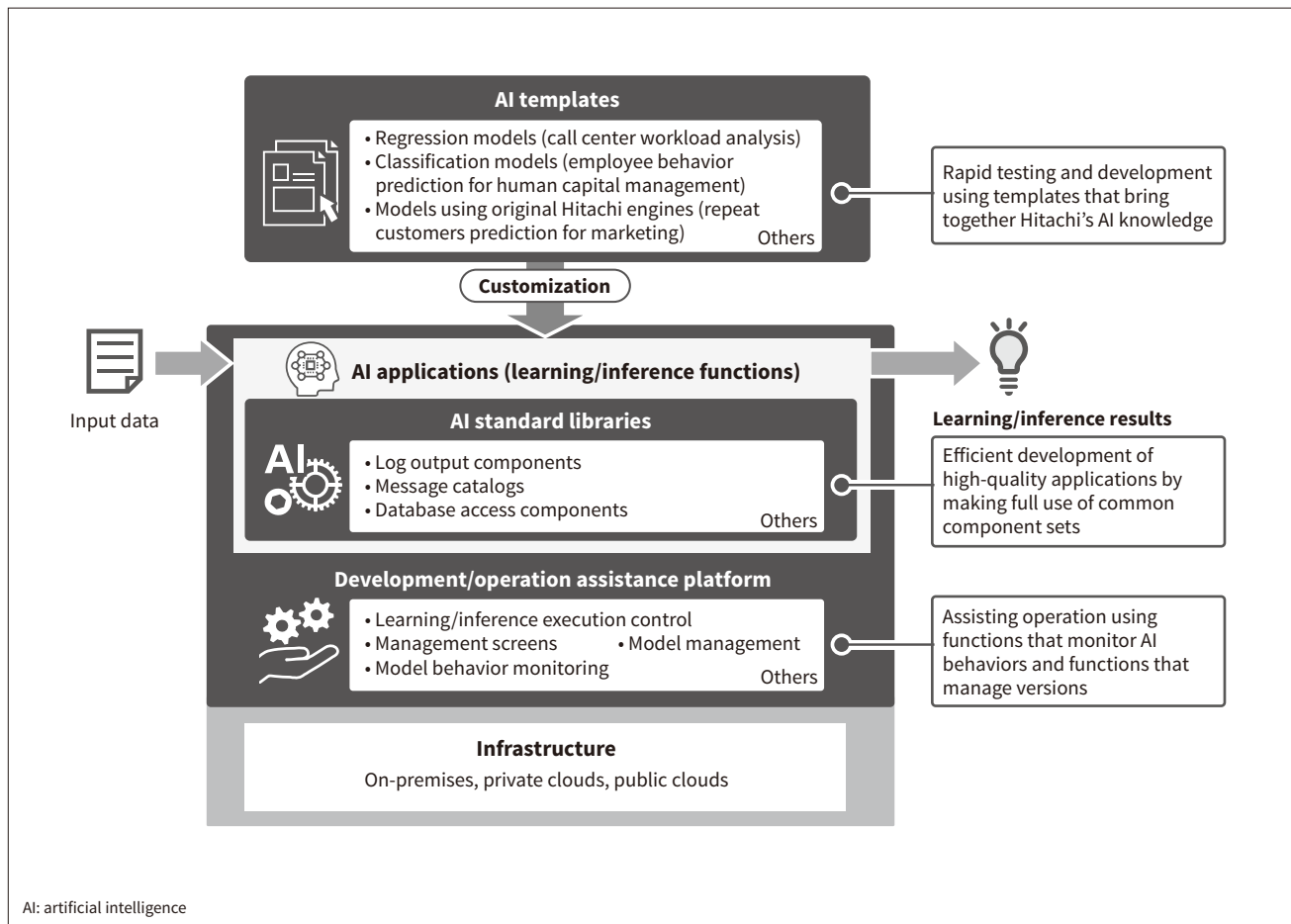
Up to now, Hitachi has developed and released a Java<sup>\*1</sup> development framework for assisting enterprise application-building known as the Hitachi Application Framework. By drawing on the enterprise system development expertise accumulated from this framework along with expertise gained from AI projects the company has worked on in a wide range of industries, a new framework has now been developed. Known as the Hitachi AI Application Framework (“the Product”), it assists AI application development for the enterprise domain and was released in April 2021 (see **Figure 1**).

This article describes the issues involved in the use of AI for enterprise systems, along with the functions that the Product provides to solve these issues.

<sup>\*1</sup> Java is a trademark or registered trademark of Oracle Corporation and its subsidiaries and affiliates in the US and other countries.

**Figure 1 — Hitachi AI Application Framework Overview**

The Product is an application framework assisting AI development and operation. It provides three functions: AI templates, AI standard libraries, and a development/operation assistance platform.



## 2. AI Applications in Enterprise Systems

The characteristics of AI applications were expected to create the issues below when adding them to enterprise systems.

### 2.1

#### Determining Benefits through Trial Development

The use of the agile development approach for developing enterprise systems is gradually starting to gain ground. Meanwhile, waterfall development is more commonly used. This approach uses the defined requirements to progressively subdivide functions, adding more and more detail as development progresses. However, defining the requirements ahead of time is very difficult for AI application development as the type of learning results returned by the AI engine will depend on the input data and parameters.

So when developing AI applications, the requirements are often set by first holding a trial development phase to verify feasibility. This phase consists of repeated cycles of learning and verification to determine the benefits of using AI. System development is then expected to begin once

these benefits are understood and the requirements have been redefined.

The use of this different approach is what makes AI applications often take longer to develop than standard applications.

### 2.2

#### System Integrity Requirements

Since enterprise-domain mission-critical systems have a major impact on companies and consumers, they demand a higher standard of integrity than is demanded for information systems or systems in other domains. Ensuring system reliability and stability is difficult on a system-by-system basis. The Hitachi Application Framework has improved system-building efficiency by providing Java control functions and common components used to ensure non-functional requirements such as reliability and stability.

Python<sup>2</sup> is often used for AI application development as it has a number of different mechanisms for making the work easier (such as more efficient use of AI engines and data). However, Python does not yet have sufficient functionality for implementing the non-functional requirements.

<sup>2</sup> Python is a registered trademark of the Python Software Foundation.

2.3

Accommodating AI-specific Characteristics

AI, and especially machine learning are made to learn by using an AI engine on a specific data set provided as input. The learned model obtained as the result is then applied to unknown data to make predictions. Programming an AI application usually means creating the application used to generate these predictions from the learned model.

Learned models can sometimes return impermissible results when the data entered in them has different characteristics from the data that was used during learning (such as a different distribution or inclusion of invalid values). Instead of just being a one-time occurrence, this different data can sometimes be the result of a change in preconditions (such as a change of seasons). So when this data is entered, learning may need to be re-done with a new data set.

Therefore, when integrating AI into a system, these AI characteristics need to be accommodated by having the system operation incorporate mechanisms for continuously monitoring and managing learned models and input/output data beforehand. This requirement is a major difference from conventional systems.

3. Development of Hitachi AI Application Framework

The Product has been designed to solve issues using the approaches described below.

3.1

Providing AI Application Templates

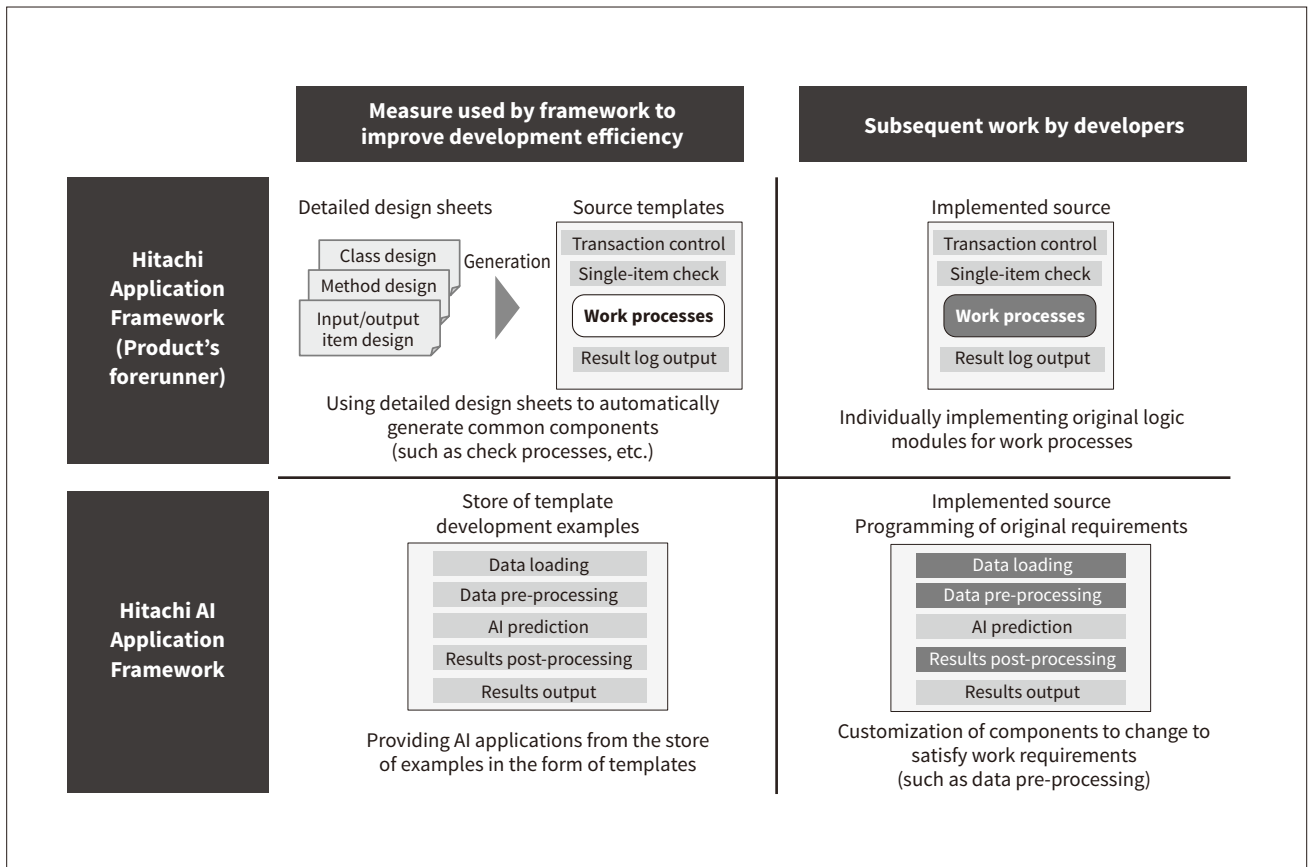
The approach to increasing application development efficiency used by the Product’s forerunner (Hitachi Application Framework) was to automatically generate source code and test code from design sheets.

However, Hitachi’s developers realized they would need to use a different approach from this automatic code generation since AI application development often includes a trial development phase before the start of design to generate results for use in the production development phase.

So the approach they opted for instead was to provide general-purpose features in the form of templates covering all the elements except for those specific to each system. These templates were derived from previous successful proof-of-concept and production AI applications (see Figure 2).

Figure 2 — Mechanisms that Improve Application Development Efficiency

The Product’s forerunner (Hitachi Application Framework) was designed to implement the original requirements by automatically generating source code templates from design sheets. The Product alters this approach by instead providing templates created from past examples and enabling them to be customized to meet original requirements.



By using these templates starting from the trial development phase and adding customizations to meet the requirements, the developers thought they would help enable faster AI application development.

To help boost AI application development efficiency, the Product’s initial release provided the following three templates:

- (1) Regression model examples (call center workload prediction)
- (2) Classification model examples (employee data-based behavior prediction)
- (3) Classification model examples using AI engines expanded by Hitachi (repeat customers prediction for marketing purposes)

### 3.2

#### Providing Common Components and Control Functions for Enterprises

Although there are some differences between Java and Python as development languages, they are not considered to have any major differences in terms of the common components and control functions needed for developing applications to be embedded in enterprise systems.

Therefore, the existing common components and control functions of Hitachi Application Framework were rebuilt for Python to efficiently fulfill the requirements for a high

degree of integrity (The common components are components for areas such as log output and message management. The control functions are used to control elements such as application execution sequences and transactions). Since the data to be used for input resides in the existing system, an interface to connect the existing system with the AI system has also been provided to enable easier use of previously gathered data (see Table 1).

### 3.3

#### Providing Mechanisms for Managing and Monitoring Learning and Prediction

As mentioned, input data that trends differently from the learned data can sometimes yield impermissible prediction results since AI applications use learned results (learned models) to run daily predictions on unknown data sets.

This characteristic led Hitachi’s developers to believe that operating a system with embedded AI applications would require the input/output data processed during learning and inference to be constantly monitored during execution, along with ongoing version management for learned models. They decided to provide these features as common mechanisms.

While monitoring requirements vary by system, it is possible to standardize the items to monitor and information to check to some extent, considering the characteristics of AI.

**Table 1 – Common Components and Control Functions for Enterprises**

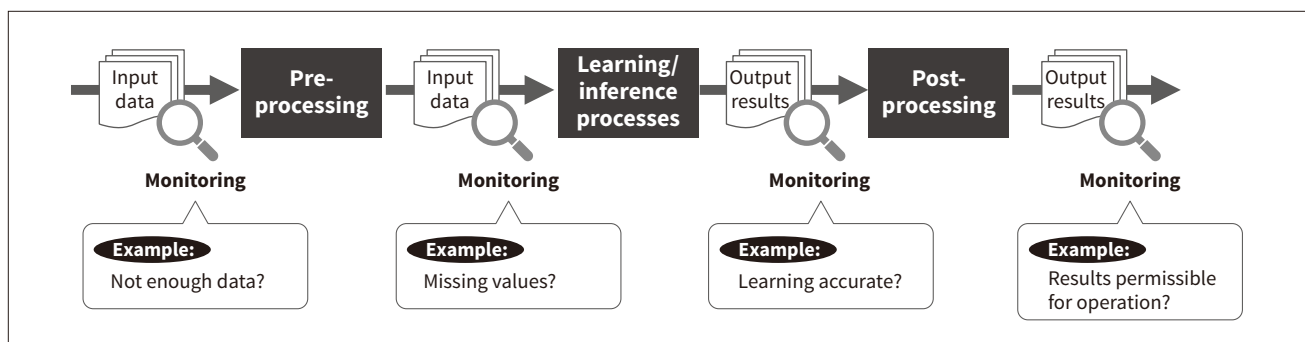
The Java common components/control functions that were provided by Hitachi Application Framework have been used to extract the essential items needed for AI application development. They are provided as rebuilt component sets for Python.

Function	Overview	
Log output components	• Log and trace outputs	
Exceptions	• Common exceptions	
Configuration	• Parameter acquisition	• Parameter caches
Message catalogs	• Message code management	• Message acquisition
Database access components	• Database connection management	• Database access assistance
CSV file linking	• Reading CSV files	• Writing to CSV files
Learning/inference execution control	• Work process call sequence control	• Transaction control
Business system coordination assistance	• API coordination assistance	• Batch startup assistance

CSV: comma-separated values API: application programming interface

**Figure 3 – AI Behavior Monitoring**

The use of preset rules to automatically monitor input data and output results allows the user to detect data changes and unexpected output.



The developers therefore decided to provide a mechanism that enables automatic error detection where users register sets of items to monitor and information to check as monitoring rules in the Product beforehand, and the Product automatically checks the input/output data processed during learning and inference (see **Figure 3**).

This mechanism was intended to enable system operation using suitable learned models without requiring any model awareness from the user. This operation is possible because the mechanism enables the Product to automate the version management of learned models and the selection of learned models used during inference.

## 4. Future Outlook

The Product's AI templates are the feature with the greatest impact on reducing application development time. When a template that can properly match the requirements to implement is available, it can be customized and quickly programmed into the system as an AI application instead of having to develop that application from scratch. This is what makes AI templates so effective, so expanding these templates will be an essential requirement for promoting AI throughout the enterprise domain. Hitachi's Lumada Data Science Lab. is leading the company's continual efforts to build up its varied portfolio of AI expertise. This expertise will be effectively applied toward continuous AI template expansion.

Since the Product is provided in the same format as a conventional software product, the user will need to do some setup work when using the Product for development or when installing applications in the system environment. Hitachi has released a platform known as Lumada Solution Hub that improves the efficiency of processes to enable digital transformation, making it easier to select and install the required solutions. Using the Product through Lumada Solution Hub will help make installation easier.

## 5. Conclusions

This article has described what Hitachi believes to be the salient issues when adding AI to the enterprise domain, along with the functions that the Hitachi AI Application Framework provides to solve these issues.

Hitachi is confident that use of this Product and its various related solutions will help overcome the obstacles facing companies who want to adopt AI. It will also help accelerate the use of AI for business.

## Reference

- 1) Hitachi, Ltd., "Hitachi AI Application Framework" in Japanese, <https://www.hitachi.co.jp/products/it/appsdiv/service/justware/ai-apfw/>

## Authors



### Tomonori Nakamura

Architecture Design Department, Architecture Design Operation, Applications Services Division, Services & Platforms Business Unit, Hitachi, Ltd. *Current work and research:* Development of Hitachi AI Application Framework.



### Yoshitaka Terada

Architecture Design Department, Architecture Design Operation, Applications Services Division, Services & Platforms Business Unit, Hitachi, Ltd. *Current work and research:* Development of Hitachi AI Application Framework.



### Chihiro Wada

Architecture Design Department, Architecture Design Operation, Applications Services Division, Services & Platforms Business Unit, Hitachi, Ltd. *Current work and research:* Development of Hitachi AI Application Framework.



### Toshiyuki Ukai

AI Business Development Department, Lumada CoE, Services & Platforms Business Unit, Hitachi, Ltd. *Current work and research:* Development of AI Framework by systematizing proven AI expertise for accelerating AI-using business. *Society memberships:* The Information Processing Society of Japan (IPJS).