# Platform to Accelerate Utilization and R&D of AI Technologies

● Kenichi Kobayashi ● Nobuyuki Kuromatsu ● Hiromichi Kobashi
● Haruyasu Ueda

**In recent years, AI technologies have been applied to various fields. At the same time, the difficulty in appropriately selecting AI technologies on the basis of actual business issues and implementing them in society is also becoming clear. In addition to the provision of reliable AI solutions come security issues and other new problems. The key to solving these issues is co-creation, where a company with a wealth of business knowledge suited for business issues cooperates with another company that creates AI technologies to explore possible solutions. Fujitsu Laboratories has developed AIEcosystem, a platform that closely connects sites that create AI technology with business sites to facilitate quick provision of AI solutions. AIEcosystem is characterized by its capability of storing AI technologies and past case examples in forms that actually function as knowledge and holding trials immediately without the need for implementation and deployment. This paper provides a technology outline of AIEcosystem and discusses the platform's usefulness.**

## 1. Introduction

In parallel with the remarkable evolution of AI in recent years, AI is being increasingly accepted by society, and opportunities for implementing it with real benefits to society are appearing at an explosive rate. Nevertheless, it has become clear that a variety of complex and difficult problems brought on by the evolution of AI technology must be dealt with if AI is to be successfully introduced into society.

For example, while the ability to solve existing business problems through the development of new AI technologies can create major business opportunities, finding the AI solution most suitable to a certain business problem from among many and varied AI technologies can be difficult. At the same time, AI technology can be highly diverse in nature since many of its components such as data processing, machine learning, and distributed processing are intricately entwined. This increases the labor required for implementing and deploying the technology and the work hours for holding trials. Additionally, as the training data input to AI increases in value, new problems such as data security, protection of personal information, and fairness are becoming increasingly complicated making it difficult to provide high-reliability AI solutions (products and services).

Given this state of affairs, Fujitsu sees its mission as helping its customers solve their business problems through the speedy and high-reliability provision of AI solutions in line with business opportunities. The key to accomplishing this is "co-creation" in which a customer as a company faced with business problems and Fujitsu as a developer of AI technology work together to create new value. Here, shortening the trial cycle between the customer and Fujitsu and increasing the frequency of interaction are essential to raising the quality of co-creation. It is also important to deal with new complexities in the world by adding new Fujitsu high-reliability technologies while appropriating past successful examples of using AI that have been accumulated as knowledge and making use of existing high-reliability implementations.

Fujitsu Laboratories has developed AIEcosystem as a new platform for supporting the provision of AI solutions in response to these needs.

This paper provides a technology overview of AIEcosystem and discusses the platform's usefulness.

## 2. AIEcosystem

This section describes the concept of the AIEcosystem platform plus its features, architecture, and use cases.

### 2.1 Concept

The concept of the AIEcosystem is to provide a platform for accumulating and running "executable examples." An executable example is a formulation of information that enables a variety of examples involving data analysis, AI solutions, the use of AI technology, etc. to be reproduced as an execution environment through the use of a Docker[1] container (hereafter, container), a lightweight virtual environment. As shown in **Figure 1**, an executable example consists of documentation, core program code, and a definition of the execution environment.

We note here that the literate programming[2] style that integrates documentation and code as reflected, for example, by Jupyter Notebook[3] has been widely used in the last few years. The executable example, however, is a new concept that expands upon that approach. The following benefits can be obtained by including the execution environment definition in the target of documentation:

- Suppresses inconsistencies between the documentation and execution environment and mismatches among OS, framework, and tool versions.
- Enables storage of know-how not explicitly contained in the documentation and of procedures like infrastructure operation.
- Enables easy reuse of execution environments for actual operation and makes it easy to combine multiple execution environments.

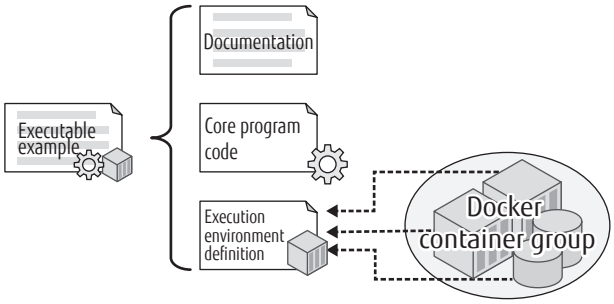


**Figure 1**
**Executable example.**

An executable example is easy to describe. Notation that users are familiar with such as Jupyter Notebook can be used to describe documentation and code, and complex elements such as distributed execution management are not needed to describe the execution environment definition.

The AIEcosystem is outlined in **Figure 2**. The user searches for an executable example matching current objectives from among the collection of stored examples, implements and deploys the example chosen, and uses it for analysis, appropriation, extension, etc. Within the AIEcosystem is a repository of executable examples that includes a function for accumulating them. To execute an example selected by the user, AIEcosystem automatically allocates the necessary computing resources from an internally managed storage server group and computing server group mounting many CPUs and graphics processing units (GPUs) to the container group, and places the example in a state that can be used by the user. This sequence of processes is completed in only several seconds, relieving the user of the burden of implementing and deploying the executable example while also eliminating the bothersome task of securing servers.

The targets of containerization envisioning use by executable examples range from single-process programs to a set of microservices made up of multiple containers, and in actuality, include all configurations that AI solutions can take on. This makes it possible to create a deployable executable example with a configuration the same as that of a final deliverable AI

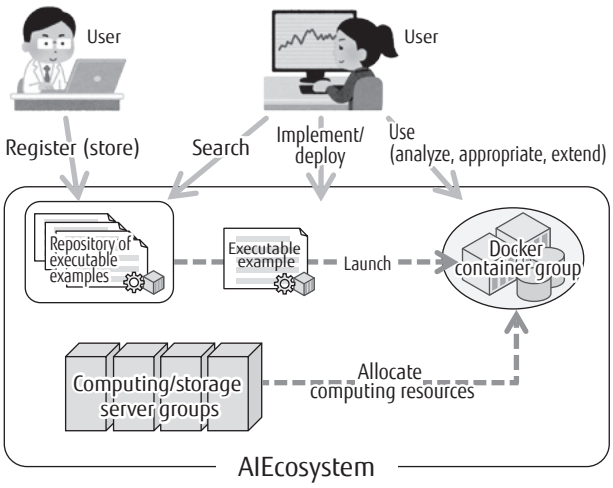


**Figure 2**
**Overview of AIEcosystem.**

solution—this is the outstanding feature of AIEcosystem.

## 2.2 Features of AIEcosystem

AIEcosystem provides the following features:

1) Enables the immediate implementation, deployment, and trials of an example on a co-creation site. A variety of practical configurations from small single programs to complex multi-server systems can be targeted for execution.
2) Enables the accumulation and searching of past examples that can serve as reference for solving business problems.
3) Simplifies the appropriation of past examples and makes it easy to extend them by adding on new AI technologies and high-reliability technologies.

To achieve these features, we established the concept of the "executable example" in AIEcosystem. In addition to being execution environments, executable examples possess properties that enable them to be stored, searched for, accessed, and customized in much the same way as documents. In other words, they take on the form of knowledge with high reusability.

The operations performed when using AIEcosystem typically result in the following flow: the user, such as a system engineer (SE) or analyst in the customer's company or in Fujitsu searches through previously stored executable examples and retrieves the one that fits current business problems, implements and deploys it, and analyzes, appropriates, and/or extends it to form a solution. In addition, the act of searching for and retrieving an executable example can quickly expose the user to that execution environment.

## 2.3 Architecture

The basic architecture of AIEcosystem is shown in **Figure 3**. This architecture is typical of a platform as a service (PaaS) infrastructure and uses open-source software (OSS), a de facto standard for software, for a number of its constituent elements. The accumulation and browsing of executable examples is achieved using the GitLab[4], the repository server of the Git[5] version control system. However, given the inefficiency of storing the massive amounts of data for AI, such as training data and models, these data are accumulated by the GlusterFS[6] distributed file system. In addition, the Kubernetes[7] container orchestration tool is used for container deployment, scaling, and management of computing resources.

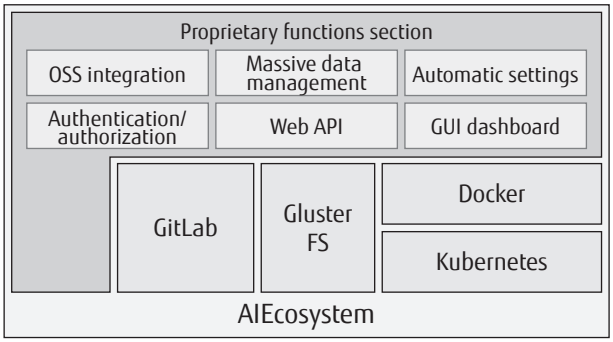


**Figure 3**
**Architecture of AIEcosystem.**

AIEcosystem includes a proprietary functions section for integrating the OSS described above and achieving the previously described features. The following describes these functions. To begin with, AIEcosystem has a function for reproducing an execution environment including massive data by mounting data stored on GlusterFS into a container when launching an executable example. Next, AIEcosystem has an automatic-settings function. In the past, users who wished to appropriately allocate computing resources and deploy containers have had to understand certain complex concepts to make settings in Kubernetes. AIEcosystem hides the complexity of Kubernetes and automatically makes settings thereby making it easier for users to use this tool.

AIEcosystem also has an authentication/authorization function for managing access rights to executable examples, data, etc. Additionally, it has a straightforward web application programming interface (API) that conceals the inner workings of OSS to make it easy for users to use the above functions, as well as a dashboard-type GUI that a user can begin using without incurring any training cost. We refer the reader to another paper written by the authors for details on the above functions.[8]

## 2.4 Use cases

We here introduce six typical use cases for AIEcosystem and illustrate, in particular, those of creating executable examples in **Figure 4**. Note that the agent in use cases 1) to 5) is the user (an SE or analyst in the customer's company or in Fujitsu).

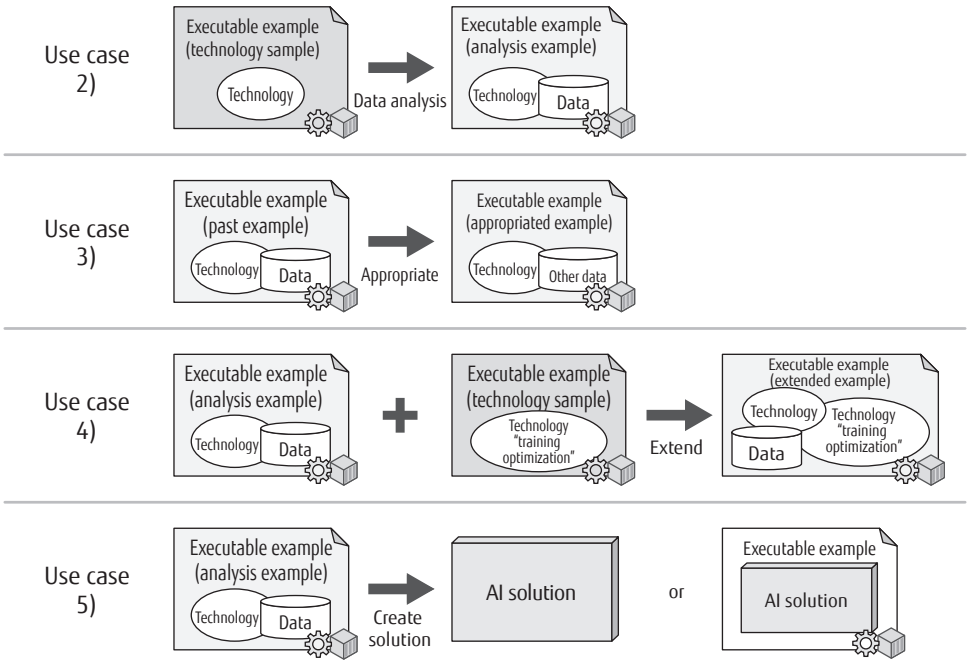


**Figure 4**
**Use cases of creating executable examples.**

1) Search for AI technology that fits business problems or a similar executable example from the past.
2) Perform data analysis using an executable example of AI technology that fits business problems and create an executable example of an analysis example.
3) Appropriate an executable example of a past example similar to business problems and create an executable example of an appropriated example.
4) Create a new extended executable example by combining a base example with an executable example of new AI technology.
5) Create a solution based on a previously created analysis example. This AI solution may be implemented as an executable example in AIEcosystem at this time.
6) Register and share analysis examples created by users and executable examples of AI technology created by Fujitsu researchers and developers in AIEcosystem.

These six use cases cover most of the use and accumulation cycle of executable examples.

# 3. Evaluation

This section evaluates the usefulness of AIEcosystem in terms of usage as well as work hours and trial period.

## 3.1 Expanded usage

AIEcosystem was first launched internally Fujitsu in FY2018. The trends in usage conditions in terms of the number of in-house users and number of registered executable examples are shown in **Figure 5**. Interviews that we conducted revealed that there were initially many users who began using AIEcosystem for its practical aspects such as ease of deployment and ease of allocating computing resources, but as time went on, the number of users and organizations using executable examples also increased. It was expected that using AIEcosystem would have great advantages, so the number of registered executable examples increased in parallel with an increase in the number of users. On the basis of the above, it can be said that the usefulness of AIEcosystem has been confirmed and that its use has been expanding.

## 3.2 Reduced work hours and shortened trial period

We here introduce the case of deploying the

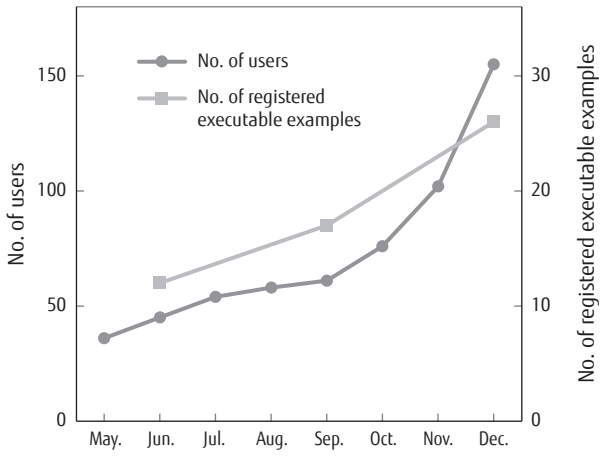


**Figure 5**
**Trends in usage of AIEcosystem (2018).**

Baker project as an example of how AIEcosystem can be used to reduce work hours and shorten the trial period. Baker is a European AI project in Fujitsu for predicting the accuracy of negotiations and analyzing risk in negotiation. We used AIEcosystem to deploy Baker in Japan.

To begin with, a version of Baker that was currently in use by a team at Fujitsu Laboratories of Europe Ltd. was registered in AIEcosystem in the form of an executable example, which was then appropriated by a team at Fujitsu Laboratories in Japan. In this way, a Baker execution environment could be implemented and deployed with one click on AIEcosystem, that is, the original environment could be reproduced instantly.

By reconstructing the environment instantly, we were able to shorten the trial period. We note here that differences in negotiation management systems and business customs between Europe and Japan resulted in a number of differences in data items, so some data preprocessing and redesign of feature quantities were necessary. However, as a result of using AIEcosystem, there were many sections that could make use of knowledge within the original Baker executable example, so these tasks could be completed with a minimum number of work hours. In the end, what took nine months to develop in Europe could be completed in two months or less in Japan.

## 4. Related technologies

This section introduces technologies similar to AIEcosystem and explains why AIEcosystem is a superior platform.

Google Colaboratory[9] is a service that provides users with a simple execution environment for Jupyter Notebook to facilitate collaborative data analysis. It integrates documentation and code via Jupyter Notebook in the same way as many other related technologies. Google Colaboratory places importance on collaborative work, a stance that shares similarities with AIEcosystem that emphasizes co-creation. AIEcosystem improves upon this by integrating the execution environment in addition to documentation and code through the use of executable examples.

Google AI Hub[10] provides a function for accumulating knowledge and enables the reuse of code and documentation as knowledge. Amazon SageMaker[11], is an integrated development environment that enables users to easily build machine-learning models on the cloud. Google AI Hub and SageMaker enable the accumulation and reuse of knowledge in the form of training jobs that do not, however, include knowledge of the execution environment. AIEcosystem, on the other hand, also enables execution environments to be reused as knowledge.

Similar to SageMaker, Microsoft Azure ML Studio[12] is an integrated development environment that can be used to build machine-learning models on the cloud with ease, and enables the use of Jupyter Notebook. However, deployment formats are predefined, which makes it difficult to build AI solutions with flexible system configurations. AIEcosystem, on the other hand, can handle execution environments with a variety of configurations from a single container to a multi-server system consisting of multiple containers.

Finally, Polyaxon[13] is a platform for building, training, and monitoring AI solutions. This platform is built upon Kubernetes and can handle flexible configurations of containers. However, it cannot use Jupyter Notebook and its training library is fixed, which places enormous burdens on the user at the time of analysis or implementation. With AIEcosystem, on the other hand, there are, of course, no limitations on the use of Jupyter Notebook or library configurations

## 5. Conclusion

This paper introduced Fujitsu's AIEcosystem as a new platform for quickly building and rolling out AI

solutions that use AI technology.

AIEcosystem gathers together AI elemental technologies and high-reliability elemental technologies to provide developers with options in choosing technologies for building AI solutions. It also accumulates successful data-analysis examples and AI solutions to enable the reuse of past examples through appropriation, extension, etc. In addition, the prompt holding of trials through the use of executable examples can increase the frequency of interaction between the customer and Fujitsu and accelerate the business implementation and rollout of AI solutions. Finally, the high degree of freedom of the AIEcosystem platform makes it easy to link to a variety of advanced technologies.

Going forward, our plan is to incorporate additional technologies such as continuous integration and continuous deployment (CI & CD) of AI systems, data privacy protection, trail management, and fairness assurance to continue the integration of AI technologies and accumulation of successful examples. In this way, we aim to enable even faster provision of high-reliability AI solutions in line with the business problems of Fujitsu's customers.



## References

1) Docker.
*https://www.docker.com/*
2) D. E. Knuth: Literate Programming. Stanford, California: Center for the Study of Language and Information 1992.
3) Project Jupyter.
*https://jupyter.org/*
4) GitLab.
*https://gitlab.com/*
5) Git.
*https://git-scm.com/*
6) Gluster.
*https://www.gluster.org/*
7) Kubernetes.
*https://kubernetes.io/*
8) N. Kuromatsu et al.: A container platform that enables quick building and applying machine learning system. 1st Machine Learning in Science and Engineering Workshop (MLSE2018), July 2018. (in Japanese)
*https://drive.google.com/file/d/18gWSSoBSr-btr9Ka5zpKMd3EKEnZKFbp/view*
9) Google Colaboratory.
*https://colab.research.google.com/notebooks/welcome.ipynb?hl=en*
10) Google AI Hub.
*https://cloud.google.com/ai-hub/*
11) Amazon SageMaker.
*https://aws.amazon.com/sagemaker/?nc1=h_ls*
12) Microsoft Azure Machine Learning.
*https://azure.microsoft.com/en-us/services/machine-learning/*
13) Polyaxon.
*https://polyaxon.com/*

**Kenichi Kobayashi**
*Fujitsu Laboratories Ltd.*
Mr. Kobayashi is currently engaged in research of AI automation and AI quality.

**Nobuyuki Kuromatsu**
Mr. Kuromatsu was engaged in research of AI platforms at Fujitsu Laboratories up to July 2019 prior to his retirement.

**Hiromichi Kobashi**
*Fujitsu Laboratories Ltd.*
Dr. Kobashi is currently engaged in research of AI automation and AI quality.

**Haruyasu Ueda**
*Fujitsu General Ltd.*
Mr. Ueda is currently engaged in development of AI technology for air conditioning equipment.