

PaaS Platform Based on Cloud Foundry

Ken Ojiri, Noburou Taniguchi, Takahiko Nagata, Shinichi Nakagawa, and Yudai Iwasaki

Abstract

The NTT Software Innovation Center is working on the development and commercialization of Cloud Foundry, which is open source PaaS (Platform as a Service) platform software. This article introduces the latest trends with Cloud Foundry and our activities related to its expansion to cloud businesses.

Keywords: cloud, open source, PaaS

1. Overview of PaaS

The NTT Software Innovation Center is developing a Platform as a Service (PaaS) platform^{*1} using Cloud Foundry [1, 2] as part of efforts to build a cloud infrastructure using open source software.

The objective is to provide a platform for an application run-time environment, that is, a software stack that combines the various types of software needed to run applications such as the operating system (OS), various libraries, frameworks, and middleware such as web servers, databases, and load balancers on top of Infrastructure as a Service (IaaS) (Fig. 1). A PaaS provider provides PaaS users with a common run-time environment that can be shared among various types of applications, as well as services such as a health monitor for running applications.

The greatest advantage of using PaaS is that it eliminates some of the work involved in building the run-time environment for application development and operations. Application development can be accelerated because test environments can be obtained easily using PaaS. Applications can be operated efficiently because computing resources with a pre-built environment can be procured easily while scaling the applications to a large size.

The need for PaaS has increased recently, and as a result, many commercial PaaS services have appeared on the market, including the Cloud[®] PaaS [3] service introduced by NTT Communications in December,

2012, as well as Google App Engine, AWS (Amazon Web Services) Elastic Beanstalk, Microsoft Azure, Heroku, and IBM Bluemix (see Table 1).

2. Cloud Foundry: open source software project

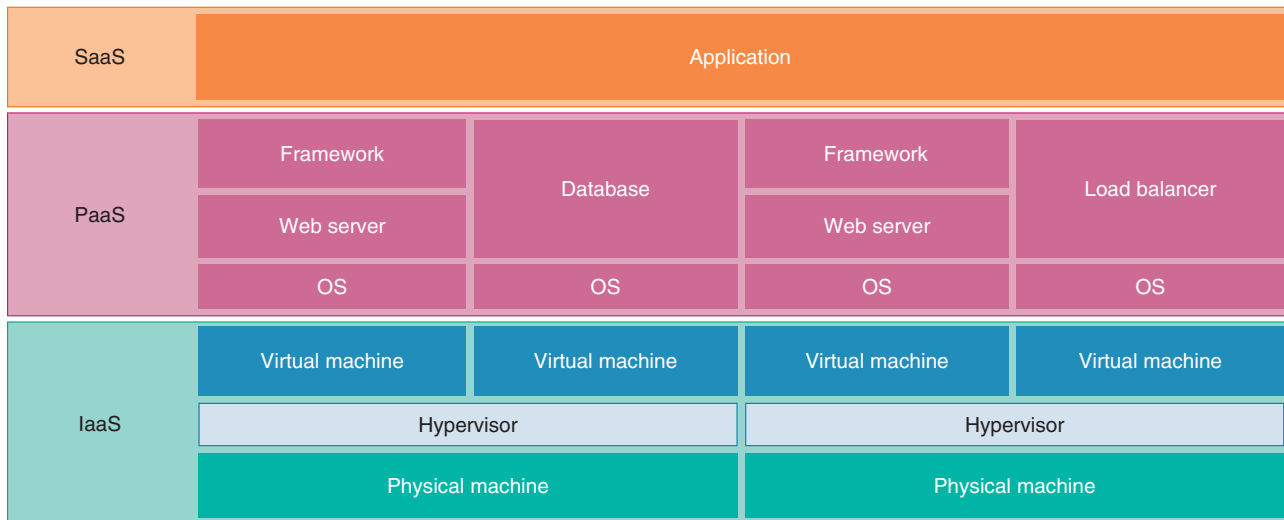
Cloud Foundry is PaaS platform software being developed as open source. In contrast to services such as Microsoft Azure or Heroku, which are built with proprietary, closed-source software, Cloud Foundry uses published source code that anyone can use to build their own PaaS platforms.

The strengths of Cloud Foundry can be summarized in three main points, described below.

(1) No vendor lock-in

Because Cloud Foundry is open source, PaaS users can avoid dependence on a particular PaaS provider (vendor lock-in). Generally, the functions provided by each PaaS and how they are used differ, so there is no compatibility between them. Because of this, migrating an application developed for a particular PaaS to another PaaS requires rewriting the source code and revising operational procedures in most

^{*1} PaaS platform: In this article, “PaaS” refers to the PaaS service itself, and “PaaS platform” refers to the technology stack used to build the PaaS. “Application” refers to applications (services) running on the PaaS. “PaaS user” refers to users developing and providing applications on the PaaS, while “application user” refers to users of those applications.



SaaS: Software as a Service

Fig. 1. Hierarchical structure of technologies comprising a cloud.

Table 1. List of PaaS services.

Service name	PaaS provider name	Features
Google App Engine	Google	Guarantees high scalability with proprietary APIs, BigTable, and other mechanisms
AWS Elastic Beanstalk	Amazon	Supports linking with other services provided using AWS
Microsoft Azure	Microsoft	Rich linking with Microsoft integrated development environment
Heroku	salesforce.com	Pioneering PaaS service supporting a variety of general programming languages
Force.com	salesforce.com	Rich business application logic
Cloud ⁿ PaaS ²	NTT Communications	Equipped with Cloud ⁿ RDB integration, logging, monitoring, distributed applet functions
Pivotal Web Services ²	Pivotal	Operated by Pivotal, leader of the Cloud Foundry community
IBM Bluemix ²	IBM	Rich applet developer support functions, linked services
anyines ²	anyines	Uses 100% European datacenters
OpenShift	Red Hat	Service using OpenShift Origin from Red Hat

² Services using Cloud Foundry

API: application programming interface
RDB: relational database

cases. This makes it difficult to handle conditions such as sudden increases in user fees or the shut-down of a service. These can pose a great risk, particularly for users who operate commercial applications. For PaaS services that use Cloud Foundry, compatibility is guaranteed and there is no vendor lock-in, so these sorts of migration problems are avoided. Users can select a PaaS provider according to conditions, or even build their own private PaaS.

(2) Flexible system configuration

To enable Cloud Foundry to satisfy varying PaaS

requirements, the overall system is divided into loosely connected components able to operate in a multiple-active configuration (n-ACT) and designed to be variously combined as needed (Fig. 2). This makes it possible to build a PaaS according to the required scale and reliability, from a small-scale, private PaaS with all components running on a single machine, to a large-scale, public PaaS operating on several thousands of machines. Components can also be added or removed afterwards, so it is easy to start by building on a small scale and later scaling out as processing requirements increase.

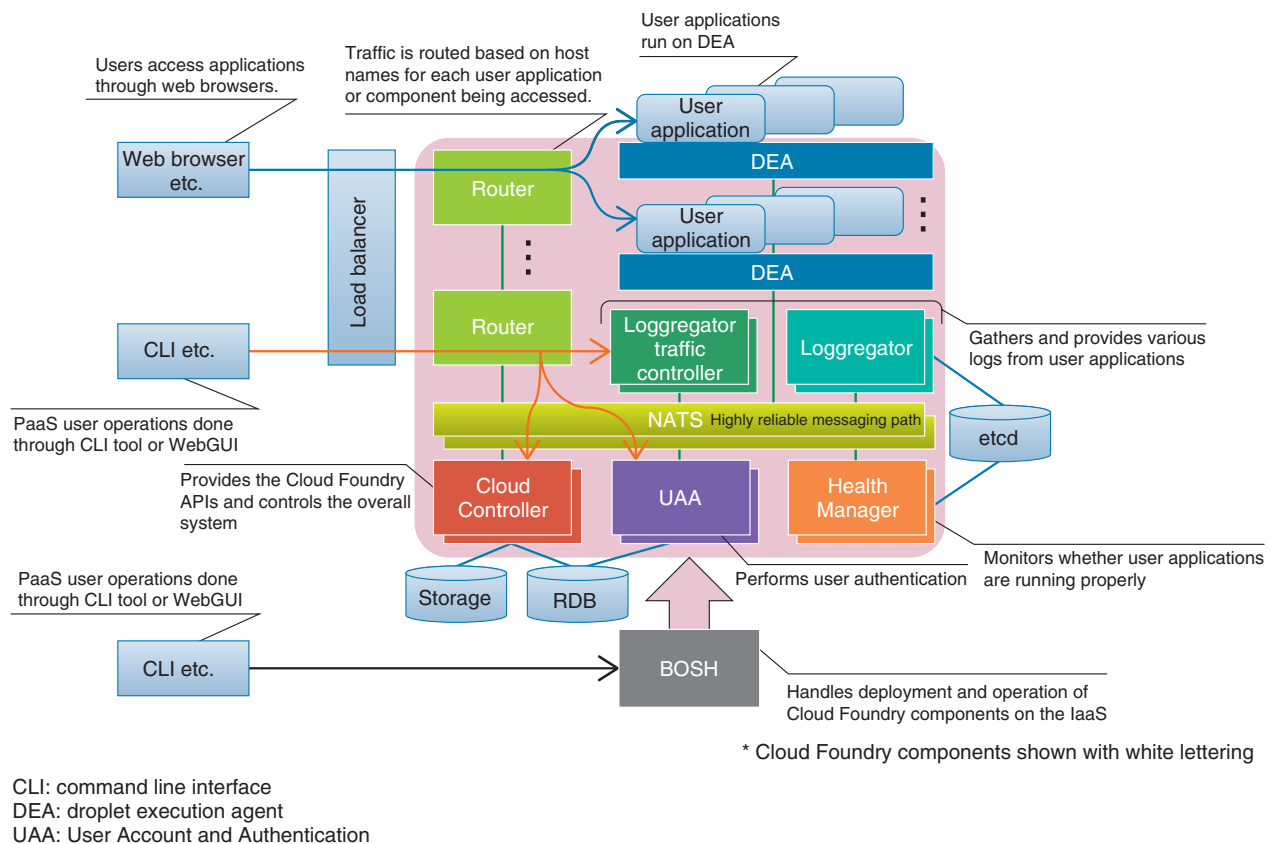


Fig. 2. Cloud Foundry component architecture.

(3) Diversity of development languages and frameworks

Applications developed using a variety of programming languages such as Ruby, Java, JavaScript, and PHP, and web application frameworks such as Ruby on Rails, Sinatra, Spring, and Node.js can run on Cloud Foundry. It also supports the Buildpack concept introduced by Heroku, which can be used to run applications using programming languages and frameworks not supplied by the PaaS platform.

Besides Cloud Foundry, there is another open source PaaS platform called OpenShift Origin, developed mainly by Red Hat Inc. At this point, Cloud Foundry is a step ahead of OpenShift Origin in terms of the number of contributing technologists, the level of activity as an open source community, and adoption by PaaS providers.

3. NTT laboratories initiatives

NTT laboratories have taken the following initiatives concerning the use of PaaS platforms based on

Cloud Foundry.

3.1 Feature expansion and verification

3.1.1 Increasing availability of applications running on the PaaS platform

When PaaS users want to ensure availability to run multiple applications, it is desirable to distribute application deployment locations as widely as possible. Therefore, we have extended Cloud Foundry application management features so that application placement can span multiple datacenters and servers.

3.1.2 Automating operations of the PaaS platform

BOSH is software for PaaS platform management. It deploys, launches, and monitors Cloud Foundry components on IaaS platforms. We have developed auto-scaling functions that cooperate with BOSH to automatically adjust the number of virtual machines on IaaS platforms when there are shortages or excesses of necessary resources such as the CPU (central processing unit), memory, and network bandwidth.

3.1.3 Building and operating a PaaS platform spanning multiple IaaS systems

BOSH, as mentioned above, provides a CPI (Cloud Provider Interface) that supports various IaaS platforms, so Cloud Foundry can be installed on various IaaS platforms. Cloud Foundry can also be used to build and operate a single PaaS platform spanning multiple IaaS platforms if IP (Internet protocol) communication is possible between the IaaS platforms. Thus, we have demonstrated the practicality of hybrid clouds built and operated using a PaaS platform that combines multiple public IaaS platforms, or combines a public IaaS platform with a private IaaS platform.

3.2 Improved efficiency of operation processes through link with NTT clouds Develop

NTT laboratories and NTT Innovation Institute, Inc. (NTT I³) [4] are conducting a collaborative experiment with *NTT clouds Develop*, a service that supports installation of Cloud Foundry and DevOps. NTT clouds Develop is a solution for implementing the service development and operations processes known as DevOps, which have attracted attention as a side effect of the recent spread of agile software development. By combining the improvements in development and operations processes due to DevOps with the operational efficiency of Cloud Foundry, we expect to see substantial improvements in the efficiency of development processes.

The use of DevOps to tightly integrate the development and operations processes, which have previously been separate, means that services can be updated dozens of times per day, much more quickly and frequently than before. Frequent service updates are essential for incorporating end-user needs in a service, so it is a strong competitive advantage in today's market of rapidly changing needs. However, there are barriers to introducing DevOps. A wide range of tools must be mastered in order to implement DevOps effectively. The tools depend on the application, including source code version control and issue ticket tracking, and also continuous integration (CI) and server configuration management automation. However, gaining mastery of all of these tools and the know-how to combine them is not a simple matter. To overcome these difficulties, NTT clouds Develop has been designed to provide services with these various functions already integrated, so that anyone can easily start using DevOps.

Integrating Cloud Foundry into NTT clouds Develop should make service delivery even more efficient.

It was already possible to build server environments automatically with NTT clouds Develop, but specialized know-how was necessary to configure features such as those for clustering and monitoring of servers. Also, setting up servers in each environment incurred costs in terms of both time and money, and managing settings in each different environment was a major operational burden.

Introducing NTT clouds Develop greatly improved efficiency in terms of development tasks such as source code and ticket management, although service delivery, which is the final step, still required a great deal of time and effort. Integrating Cloud Foundry as a service delivery platform improved these issues significantly. With Cloud Foundry, service delivery is simplified to just setting parameters and pushing out the source code. This enables not only the dedicated operations team to perform the final updates but the developers themselves to do it, too. The time required for the pushed code to actually run can also be reduced to minutes or even tens of seconds, so time costs are also reduced. Cloud Foundry can also provide multiple environments, namely development and production, so computing resources are used more efficiently, and moving from the development to the production environment is easier. When Cloud Foundry is combined with NTT clouds Develop in this way, the integration of development and operations, which is the key strength of DevOps, can be taken to a higher level.

3.3 Contribution to the community

Having an active open-source community is essential for stability when using an open source product such as Cloud Foundry. Thus, NTT laboratories will participate in the Cloud Foundry Foundation [5] representing the NTT Group, and will work to build the Cloud Foundry community and expand PaaS in Japan. With these two initiatives, we will also actively contribute to the community by sharing information such as publishing source code and making presentations at the Cloud Foundry Summit.

4. Future prospects

Cloud Foundry is excellent open source PaaS platform software with very attractive features. NTT laboratories will continue to contribute to the Cloud Foundry community and work to expand the PaaS market. In the future, in addition to our current initiatives, we will also work to enhance links between Cloud Foundry and OpenStack, which is an influential

open source IaaS platform, and to expand the development of PaaS platforms to include private-cloud use.

References

- [1] Website of Cloud Foundry, <http://cloudfoundry.org/>
- [2] Website of Cloud Foundry, GitHub, <https://github.com/cloudfoundry>
- [3] Website of NTT Communications, Cloud[®] PaaS service, http://www.ntt.com/cloudn_e/
- [4] Website of NTT I³, <http://www.ntti3.com/>
- [5] Cloud Foundry Foundation, December 9, 2014, “Cloud Foundry Foundation Established to Advance Platform-as-a-Service,” <http://www.cloudfoundry.org/cloud-foundry-foundation-launch.html>



Ken Ojiri

Senior Research Engineer, Cloud System SE Project, NTT Software Innovation Center.

He received the B.S. and M.S. in engineering from Osaka University in 1994 and 1996, respectively. Since joining NTT in 1996, he has been engaged in research and development (R&D) of advanced intelligent networks, identity management of Internet services, and cloud technologies. He is now focusing on expanding the PaaS market and reforming the ways of Internet application service development.



Shinichi Nakagawa

Senior Research Engineer, Supervisor, Cloud System SE Project, NTT Software Innovation Center.

He received the B.S. in physics from Saitama University in 1987 and the M.S. in physics from Chiba University in 1989. Since joining NTT in 1989, he has been engaged in R&D of telephone and IP network systems, knowledge and information sharing systems, and cloud systems. He is a member of IEICE (Institute of Electronics, Information and Communication Engineers).



Noburo Taniguchi

Research Engineer, Platform Technology SE Project, NTT Software Innovation Center.

He received the B.E. and M.E. in mechanical engineering from the University of Tokyo in 1992 and 1994, respectively. Since joining NTT in 1994, he has been engaged in R&D of multimedia search engines, a copyright management system, identity management technology (including standardization activities in Liberty Alliance), and an XML data management system. During 2010–2014, he worked on a Desktop as a Service system and a PaaS system at NTT Software Corporation. He is currently involved in R&D of PaaS systems. He is a member of the Information Processing Society of Japan (IPSI).



Yudai Iwasaki

Software Engineer, NTT Innovation Institute, Inc.

He received the B.A. and M.M.G. from Keio University, Kanagawa, in 2009 and 2011, respectively. He joined NTT Information Sharing Platform Laboratories in 2009 and worked on Cloud Foundry for three years. In August 2014, he moved to NTT Innovation Institute, Inc. and is now developing advanced cloud platforms. His research interests are the Semantic Web, cloud computing technology, and the Ada programming language.



Takahiko Nagata

Senior Research Engineer, Cloud System SE Project, NTT Software Innovation Center.

He received the B.E. and M.E. from the Faculty of Instrumentation Engineering, Hiroshima University, in 1993 and 1995, respectively. Since joining NTT in 1995, he has been engaged in R&D of high-speed communication processing boards, reliable multicast delivery systems, secure electronic voting systems, secure file delivery systems, and cloud computing technology.