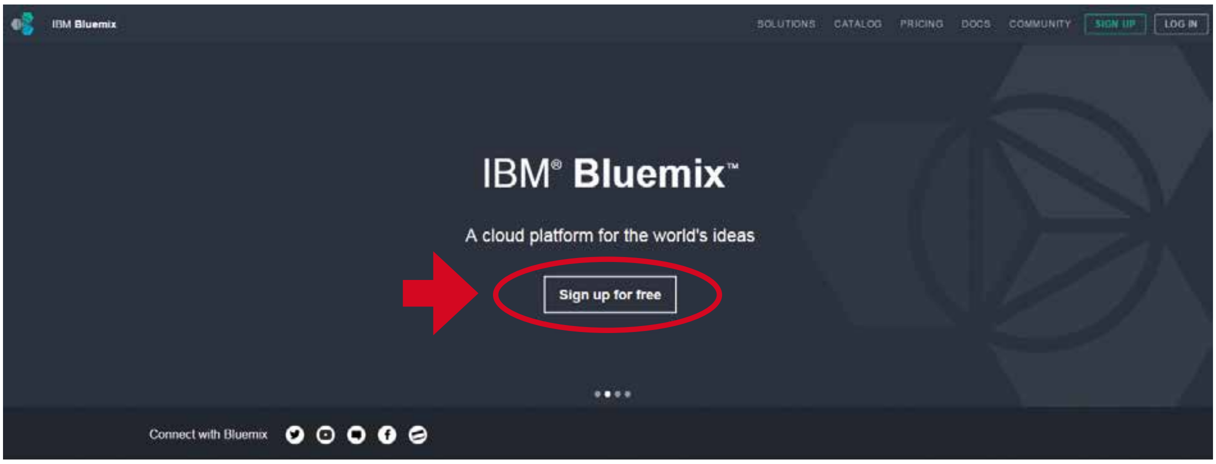
IBM® Bluemix™ From Idea to Application

Platform as a service

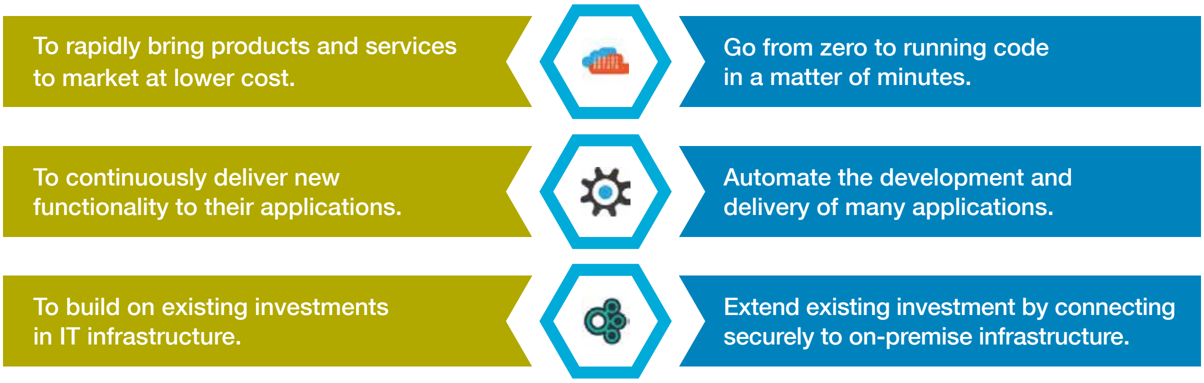
IBM Bluemix™

Before continuing, please register at Bluemix.net



With Bluemix™, you can quickly build, manage and run a range of applications (including web, mobile, big data, new smart devices, etc.) on an open-standard, cloud-based platform—free of charge within a 30-days-trial and afterwoods with half a GB of free memory every month, and tons of services with free plans—with the click on a button, and in just a few minutes.

Why are developers using Bluemix™?



To rapidly bring products and services to market at lower cost.

Go from zero to running code in a matter of minutes.

To continuously deliver new functionality to their applications.

Automate the development and delivery of many applications.

To build on existing investments in IT infrastructure.

Extend existing investment by connecting securely to on-premise infrastructure.

Contents

1. Introduction

Today's innovations are primarily based on software requiring constant delivery cycles, but twice-yearly or even quarterly release cycles are simply not sufficient. The application landscape can be divided into two segments, "System of Record" (SoR) and "Systems of Engagement" (SoE) with the SoE segment fueling innovation.

SoR applications provide the foundations for an enterprise's application landscape—including core systems that are focused on transactions, data consistency or high available resources. They bring together a company's central insurance or banking systems, HR and CRM solutions—everything that is required to support its core business. Nevertheless, there is not a huge amount of innovation within this segment of the application landscape.

They are the SoE applications where most innovation can be expected: with interaction-centric applications that embed the notions of big data and business analytics, Internet-of-Things (IoT), mobile, social and Watson. Here, interaction with customers, partners and users takes precedent. Users are most productive if they are not only supported to communicate but also well supported in their decision process. After looking at the differences between SoR and SoE, we can conclude that it is not only a case of either SoR or SoE—the focus should be on hybrid models that integrate both. SoE applications usually have to be integrated with core either or backend systems, but Bluemix is focused on developing SoE applications and integrating them into an enterprise's existing support and core systems.

This document will demonstrate how straightforward it is, to rapidly build an application in a cloud operating environment (OE) using Bluemix, as well as a GIT repository for source code management and collaboration called "IBM Bluemix DevOps Services", which makes it possible to develop and operate the whole environment in the cloud. For example, a mobile application— which is a SoE application—can connect to a company's transactional data or inventory data on-site and respond to its business needs while maintaining security, quality of service, and availability—all the things that enterprises are seeking.

Bluemix is a new platform as a service (PaaS) offering from IBM, that enables this type of integration between SoR and SoE, by combining IBM's software, third-party, and open source technologies. To simplify application design for developers, enterprises and independent software vendors, IBM also offers IBM Bluemix DevOps Services in the cloud.

This document is not only aimed at independent developers, enterprise teams, and IT architects, but at anyone with an interest in developing in a cloud OE and building apps quickly. By the end of this document, the reader should be able to understand and navigate Bluemix, use IBM Bluemix DevOps Services, and be able to develop and deploy a basic Bluemix application. A key benefit brought to the table by Bluemix is IBM's middleware portfolio, including IBM WebSphere Application Server. The middleware portfolio is exposed and predefined in IBM's cloud system, which runs on SoftLayer —IBM's infrastructure as a service (IaaS) offering—and helps deliver existing apps to the cloud. Bluemix outperforms expectations and stands out on the market, thanks to the availability of more than 100 services and middleware from IBM, its business partners, and open source software.

2. Challenges and obstacles for developers

After coming up with an idea for an application, developers face a series of obstacles throughout the development lifecycle. These obstacles could even disrupt the lifecycle process itself, consuming much time and wasting resources.

Before an application could be built, developers used to have to summarize the app's functional requirements in order to ensure the correct platform was available and running. This platform includes the hardware architecture, an operating system and runtime libraries—each of which has its own specialized administrator to install and configure it, resulting in significant downtime and delay. The specialized administrators need to set-up middleware on top of the platform—essentially an operating system—in addition to installing the correct runtime environment. If an application requires the end user to input some information, an additional database must be installed which means that, the application must be manually configured and enabled to communicate with the database, while the administrator monitors and manages the database (see Figure 1).

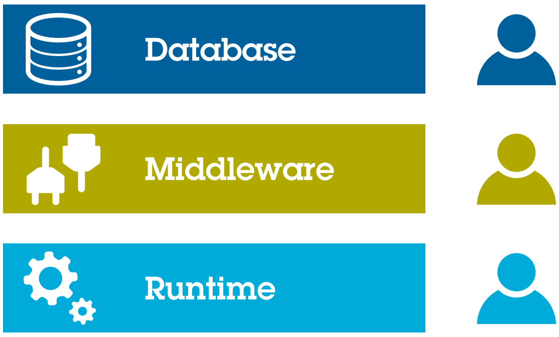


Figure 1: **Specialist administrators**

Managing and configuring secure web sockets, IP addresses or load-balancers would take much more time as-well. Once an application was finally coded and ready for use, developers then faced the challenge of releasing it into production. This also involved everything previously described around ensuring the availability of a suitable computing platform, along with setting up a middleware and runtime environment. Adding to that, the database also had to be configured on a large number of servers, which are required to withstand a much higher workload (see Figure 2). Moreover, it takes an enterprise an average of three months to purchase a new server and integrate it with their hardware infrastructure. This aspect is very important because the application could scale up rapidly at any time, and would then require a fast and reliable solution to compensate for the workload.

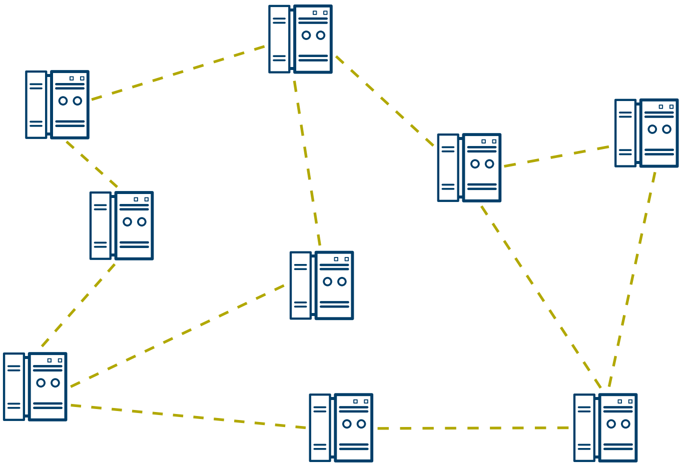


Figure 2: **A server network**

3. Positioning IBM Bluemix in the cloud landscape

3.1. Different models of cloud computing

The introduction of cloud computing provide new tools and techniques to reduce the time a developer needs to go from idea to application. What used to take weeks or months to be done, may now take only a few days or even hours. The “as a service” model has simplified developers’ workload and facilitated a transformation using the cloud. IaaS reduces the effort required to manage the hardware, while PaaS reduces the effort needed to manage the middleware and “Software as a service” (SaaS), where end users access and use business applications hosted and maintained by the vendor.

To summarize, both IaaS and PaaS cloud models simplify the workload of developers by managing complex tasks that are inferior to the developers’ core, along with important tasks and freeing up more time for them to create innovative new applications (see Figure 3).

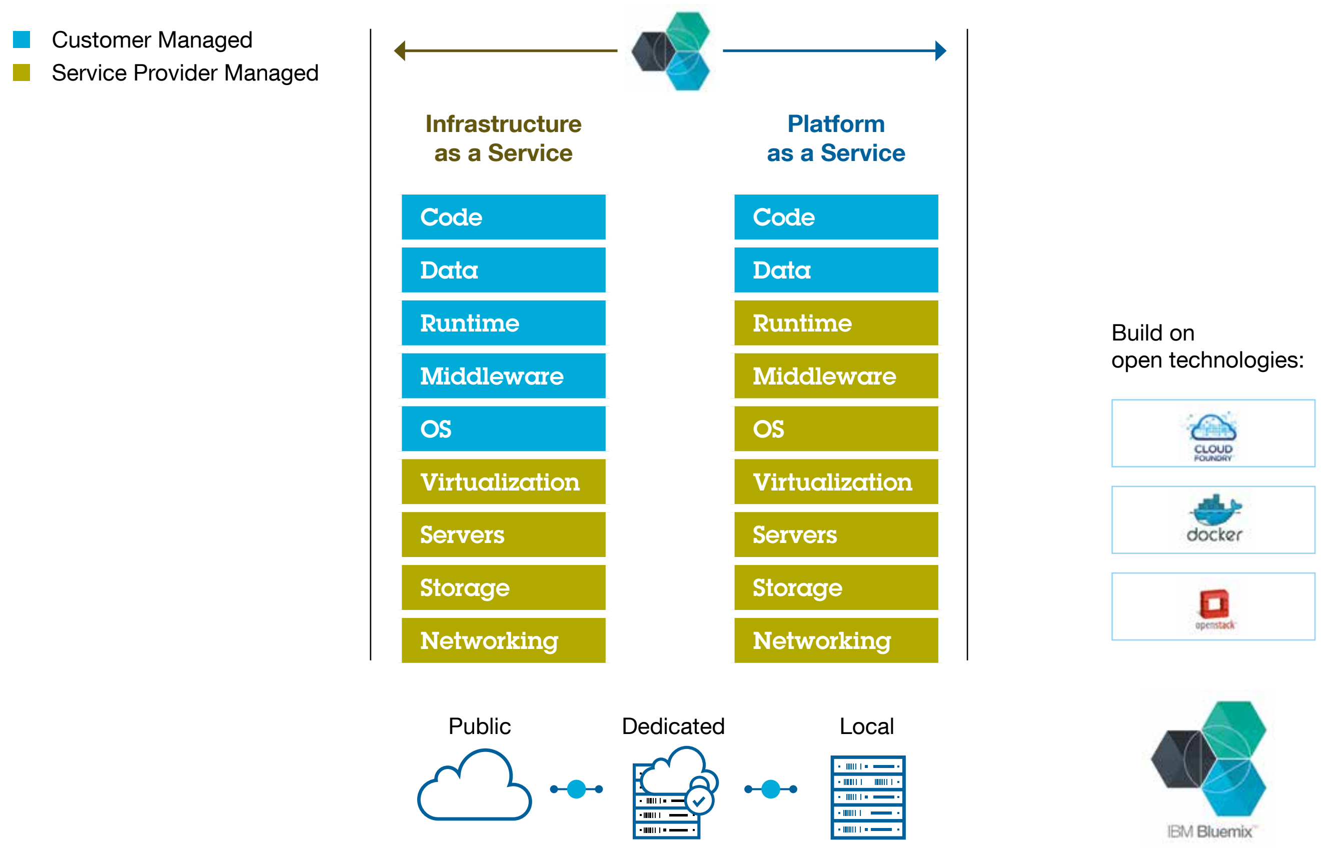


Figure 3: **Cloud computing and traditional IT architectures**

3.2. Developer-centric PaaS

From the perspective of a developer, a good PaaS offering provide a variety of runtimes and services, which makes the application development lifecycle faster and easier. In other words, when a developer considers using a platform, one of the primary things they look for is whether their favorite programming language or runtime is available— for example Java, Ruby, PHP, or JavaScript (see Figure 4). These runtimes are available through PaaS at the click of a button, so a developer can start coding immediately. Before, developers had to wait for an IT administrator to install the runtime (not the case in Bluemix) and then configure an integrated database before building the application. These tasks are all handled by PaaS, adding to the broad range of benefits it offers. The database has an allocated size—but what happens if the volume of data reaches its limit? PaaS provides a solution for this, making it possible to scale up things like runtimes or databases more granularly without the need to provision an entire VM, which in fact would lead to higher costs. Furthermore, without PaaS, these services must be installed and configured—but with PaaS, developer simply need to write a code-snippet to connect a service to your application, and then interact with the existing web or mobile application where it will be used.

Figure 4: **Programming Languages**

One of the most important features of PaaS is that it enables the real-time servicing of data—also referred to as "continuous delivery". This expression highlights an important element of the development lifecycle, which might include deploying a new coded application on the live server or modifying an existing running application with no downtime. It sums up the benefits for developers, freeing them from managing infrastructure resources and providing more time for writing code.

4. An introduction to IBM Bluemix

Bluemix is IBM's newest PaaS offering, and has been developed by a community of IBM experts working in partnership with our clients. This is one way how IBM implements its next-generation PaaS model for an open cloud architecture, by focusing on the capabilities of the cloud OE mentioned before. It enables our clients to rapidly create and deploy applications. Application environments are essentially comprised of a runtime, a set of services and a developer's own code. The services could be infrastructure, middleware, big data, mobile applications, development, and operational or other peripheral services.

4.1. IBM Bluemix architecture

Bluemix offers three open compute technologies: Cloud Foundry, Docker and OpenStack. As a result, the apps can run using instant runtimes, containers or virtual machines respectively, where you can pick the level of infrastructure for the app's architectural needs. It is enhanced with IBM's middleware (services), in addition to third-party and community-built services. There are three unique ways to deploy the apps, whether it is Bluemix Public or Dedicated—both are powered by IBM SoftLayer 40 data centers around the world—or Bluemix local which runs within your data center (see Figure 5).

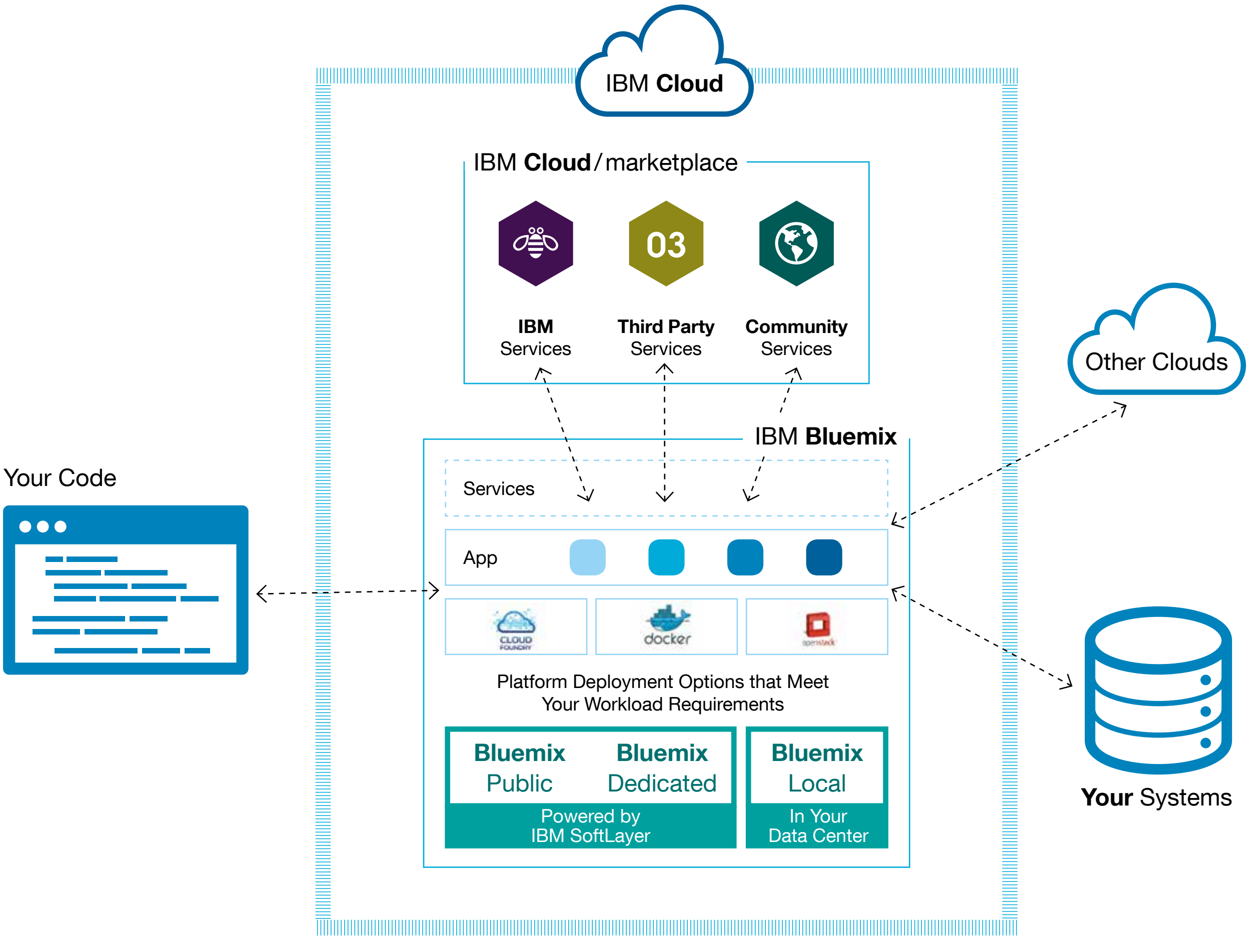


Figure 5: **Bluemix architecture**

4.1.1. Cloud Foundry

Cloud Foundry is an open-standard, open-source PaaS framework and community, funded and managed by IBM and other vendors. It runs on a broad range of cloud infrastructure platforms, including OpenStack, VMware, and IBM SoftLayer. It also supports a wide range of application programming language runtimes and frameworks, including Java, Ruby, JavaScript, and Python (see Figure 6). Cloud Foundry has a large, vibrant ecosystem of partners, vendors, and users, enabling enterprises to develop cloud-centric applications based on its open standard without fear of vendor lock-in.

What does Cloud Foundry run?

- Multi-Language
  - Ruby, Java, Scala, Node.js, Erlang, Python, PHP
- Multi-Framework
  - Rails, Sinatra, Spring, Grails, Express, Lift
- Multi-Services
  - MySQL, Postgres, MongoDB, Redis, RabbitMQ
- Multi-Cloud, Multi-IaaS
  - Public cloud, Micro cloud, Private cloud

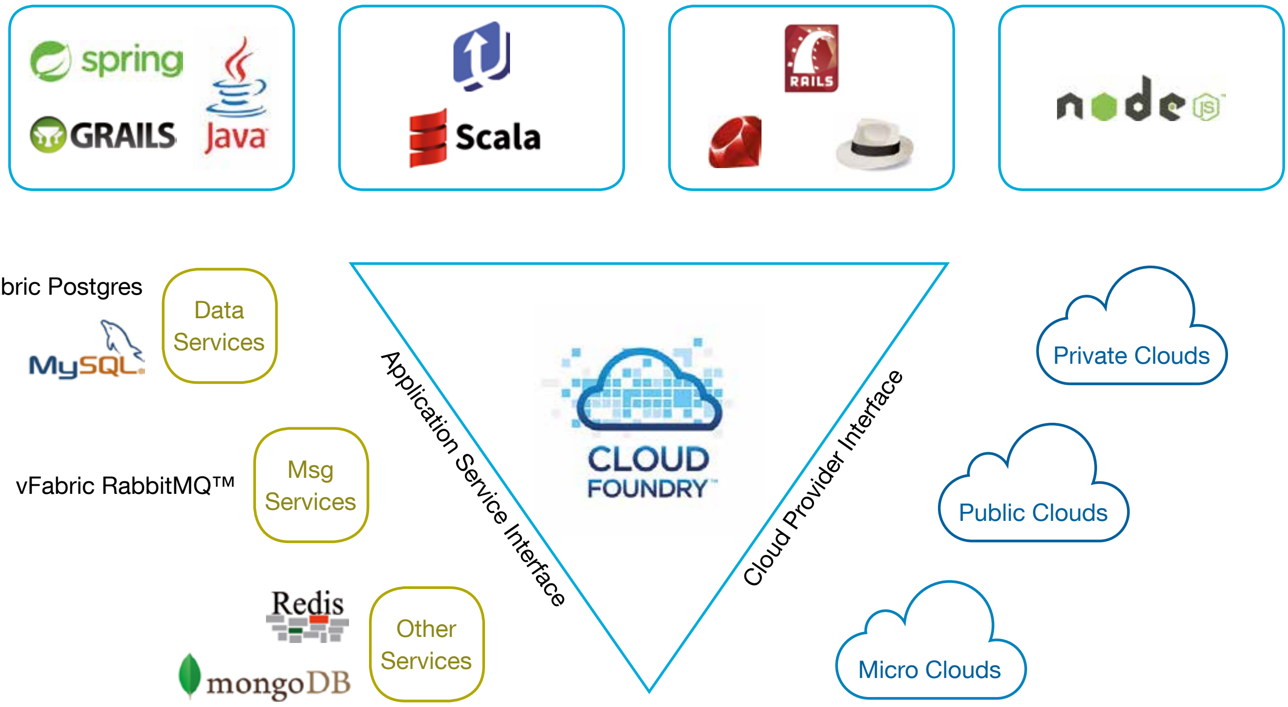


Figure 6: **Cloud Foundry**

4.1.2. SoftLayer

SoftLayer is IBM's Bluemix IaaS, providing a seamlessly unified global cloud computing infrastructure. It combines virtual public cloud instances, powerful bare metal servers, turnkey private clouds, and a broad range of storage, network and security devices and services. Since Bluemix Public and Bluemix Dedicated run entirely on SoftLayer infrastructure, it is provided by 40 data centers across North America, Europe and Asia. These data centers are intended to meet clients' growing needs in terms of choice, compliance, and data residency.

4.2. IBM Bluemix features

a) Bluemix combines the application's development with its lifecycle management. This combination is exposed across the management, testing, configuration and software maintenance by leveraging some of the capabilities of IBM Bluemix DevOps Services (see Figure 7).

b) Ready-to-use services that are created whether by IBM or our business partners which can enhance the functionality of an application.

c) Intuitive graphical user interface.

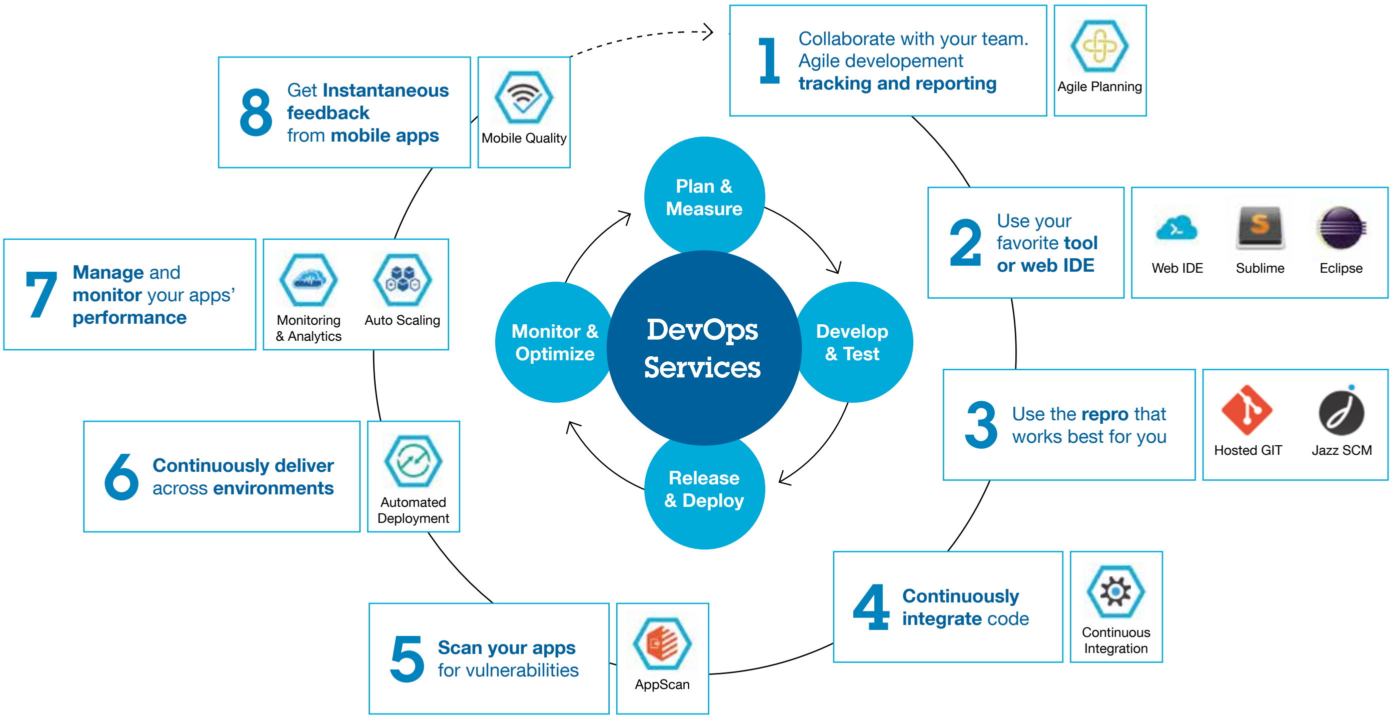
d) Free trial registration, where no credit card information is required.

e) Mix and match services and databases in a highly integrated environment and capability to scale the performance as workload increases.

f) Based on an open-source cloud computing technologies (Cloud Foundry, Docker and OpenStack).

g) Scalability through the quick provision of SoftLayer infrastructure. SoftLayer—an IBM company—provides cloud infrastructure as a service from 40 data centers in the North America, Asia, and Europe.

h) Deploy apps to Bluemix Public or Bluemix Dedicated (powered by IBM SoftLayer), or Bluemix Local that runs within your data center.



1 Collaborate with your team. Agile developement tracking and reporting
Agile Planning
2 Use your favorite tool or web IDE
Web IDE
Sublime
Eclipse
3 Use the repro that works best for you
Hosted GIT
Jazz SCM
4 Continuously integrate code
Continuous Integration
5 Scan your apps for vulnerabilities
AppScan
6 Continuously deliver across environments
Automated Deployment
7 Manage and monitor your apps' performance
Monitoring & Analytics
Auto Scaling
8 Get Instantaneous feedback from mobile apps
Mobile Quality
Plan & Measure
Develop & Test
Release & Deploy
Monitor & Optimize
DevOps Services

Figure 7: **Application Lifecycle on Bluemix**

5. IBM DevOps Services and development tools

IBM Bluemix DevOps Services is a GIT repository for source code management and collaboration in the cloud. It provides an open, integrated, rapid development experience that scales, where you can create applications using a web integrated development environment (IDE) that is associated to a GIT repository, allowing quick deployments back to Bluemix. The web IDE offers tools and processes for building mobile and cloud applications, enabling individuals and teams to plan, track, version control, develop and deploy software in the cloud. IBM Bluemix DevOps Services provides a lot of other capabilities that are useful for application development, in fact it is all about development, with features including:

- Free public project hosting
- Multiple source code repositories
- Support for teams doing Agile development
- Web-based development tools with customizations specifically for working with Bluemix
- Continuous delivery pipeline to deploy applications to Bluemix

Developers can also edit code on their local machines using a text editor or a development environment such as Eclipse, and deploy back to Bluemix through the Cloud Foundry command line interface (see Figure 8).

To modify the code, you can leverage the capabilities offered by IBM Bluemix DevOps Services powered by JazzHub. In the tutorial, we will go through the steps involved in integrating the application with IBM Bluemix DevOps Services on Bluemix.

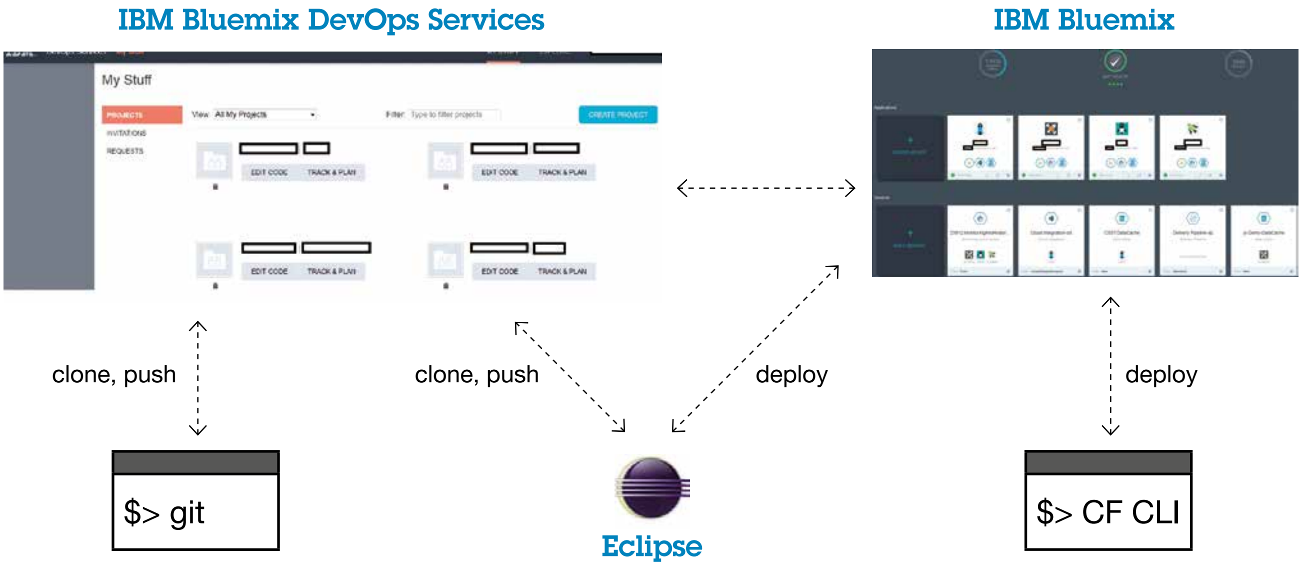


Figure 8: **Integration of Bluemix with IBM DevOps Services and development tools**

6. Security

Bluemix Public and Bluemix Dedicated are hosted on IBM's SoftLayer IaaS and therefore they benefit from the physical and infrastructure's security features like firewalls or intrusion prevention systems. Moreover, Bluemix offers a wide range of security capabilities across different categories like functional, operational and application. These categories involves authentication, authorization and auditing and furthermore user-access and automated-patch management solutions. Besides, it simplifies the management of who can sign in to cloud applications, and scan those applications for vulnerabilities with the embedded security controls into data management and big data services.

For the application category itself, Bluemix offers services like single sign-on, database security and user registry. Bluemix security policies are based on IBM IT Standards, IBM Secure Engineering best practices, and Regulatory Compliance requirements. For more detailed information about Bluemix security measures, please visit the following website:

ng.bluemix.net/docs/#overview/index-gentopic1.html

7. First steps in using IBM Bluemix (a five-minute tutorial)

This tutorial consists of three sections: At first we will go through the registration procedure in case you have not yet signed up to Bluemix. In the following section, we will build a basic web application on Bluemix using two of IBM's services on top of Node.js runtime. The app will be immediately accessible online, so it can be tested by anyone with internet access. The final section will look at how we can use IBM's Bluemix DevOps services to change the code and modify the application with flawless integration, for continuous delivery.

Tip: You can change and modify the code by 3 different ways: Eclipse plugin for Bluemix, the Cloud Foundry command line interface and IBM Bluemix DevOps Services

7.1. Registration process

a) Please go to the website Bluemix.net.

b) Click on "Sign up for a free Bluemix trial".

c) Log on with your IBM ID: If you don't have one, then register now with your work email address at **ibm.com**/account/profile/us?page=reg.

d) You will receive a confirmation note for registering to Bluemix and a secure access code by email.

e) Follow the remaining instructions, then log in with your credentials and the Dashboard will appear (see Figure 9).

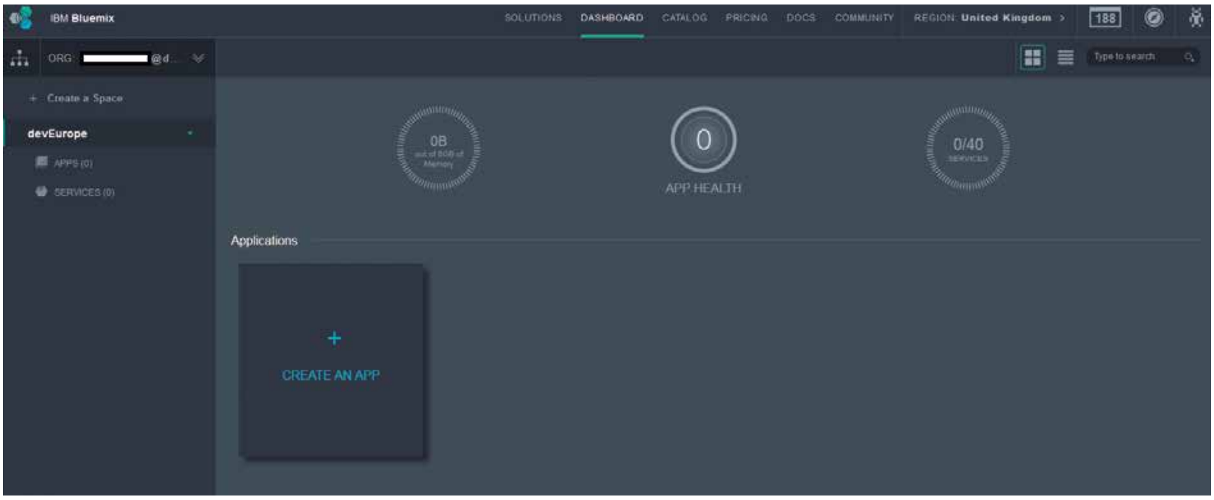


Figure 9: **Bluemix Dashboard**

7.2. Building the application

After you have registered for a free account, log in and you will be directed to the Dashboard. This is where you can create, manage, and view your application status. Moreover, it shows all the apps that are running on your account (we call your account an "Organization"), including services that were used for and bound to your app. Click on Catalog at the top, next to the Dashboard, and you will see three sections. The first is Boilerplates, which are basic, preconfigured applications defined for different purposes, and which offer a simple way to get started. If you scroll down, you will find the section with available Runtimes, which a developer can choose from based on their preference (see Figure 10).

In addition to the runtimes listed, developers are able to upload other programming languages, by selecting Bring Your Buildpack. A little further down, the third section features services that have been created by IBM itself or by IBM-certified partners such as Twilio or Pitney Bowes. Other services are community-created services, which are growing in number all the time.

Our tutorial example uses one of the boilerplates at the top of the Catalog: "Node JS Web starter". This is a Node.js runtime that uses DataCache service along with Monitoring and Analytics service on the cloud.

a) Click on the boilerplate and a window will open with more information about its functions (see Figure 11). Click on DataCache tile then "Box 1" will show you the free tier for the service. The free tier is available to all the services on Bluemix to varying degrees, and can be used to create your applications and proof of concepts free of charge.

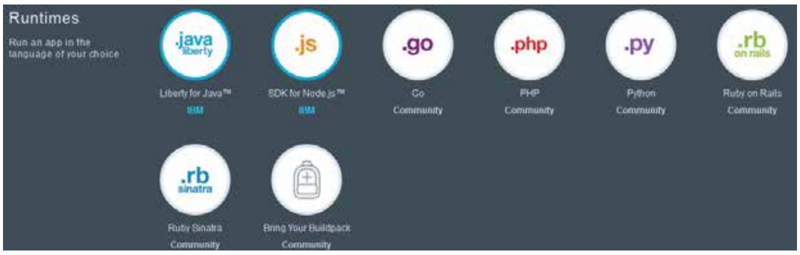
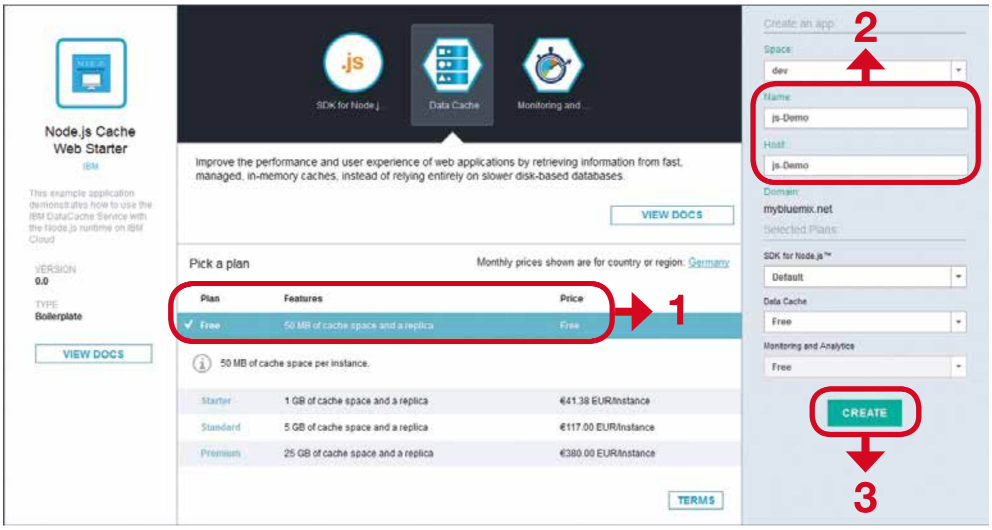


Figure 10: **Some of Programming Languages in Bluemix**



This example application demonstrates how to use the IBM DataCache Service with the Node.js runtime on IBM Cloud

VERSION
0.0

TYPE
Boilerplate

Improve the performance and user experience of web applications by retrieving information from fast, managed, in-memory caches, instead of relying entirely on slower disk-based databases.

Pick a plan

Monthly prices shown are for country or region: Germany

Plan	Features	Price
✓ Free	50 MB of cache space and a replica	Free

50 MB of cache space per instance.

Starter	1 GB of cache space and a replica	€41.38 EUR/instance
Standard	5 GB of cache space and a replica	€117.00 EUR/instance
Premium	25 GB of cache space and a replica	€380.00 EUR/instance

Create an app

Space: dev

Name: js-Demo

Host: js-Demo

Domain: mybluemix.net

Selected Plans

SDK for Node.js™: Default

Data Cache: Free

Monitoring and Analytics: Free

Figure 11: **Create an Application Window**

b) The Nameand Host-fields must be completed in “Box 2”. The application name is the name of the application on the Bluemix Dashboard, while the host is the app’s unique URL, which you use to access your application online.

c) Click CREATE in “Box 3” and Bluemix will start provisioning the Node.js runtime environment and connecting both services to the runtime.

App tile on the Dashboard

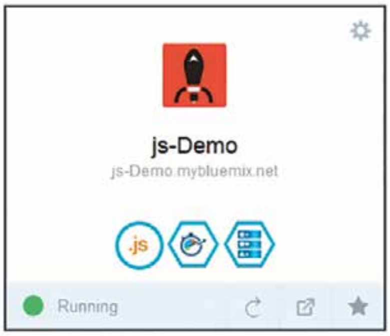
The instance is now up and running—all in around 30 seconds. The app tile will now appear on the Dashboard, showing that it is running. Without Bluemix, this step would have required me to create a virtual machine, configure the memory, configure infrastructure software, deal with routers, and allocate a domain name.

d) The application is now up and running. To open the application on the Internet, click on the Dashboard, then scroll down to find the application that you have just created, in my case it is called “js-Demo”. Click on it (see Figure 12) and then click on the URL in the top left corner next to Routes.

e) Try the online application with its basic functionalities, using “Put” to insert values and you get “Put successfully” with a red font color and retrieve them using “Get” (see Figure 13). It is a very basic application where there is a user-interface written in Node.js and there is DataCache service behind it that receives your entries and stores it for you. This service could be also replaced by a database which has the same functionality.

f) All working fine? Great. Now let’s look at how we can modify the code itself. Switch back to Bluemix and Click on Dashboard on top where you can click again on the application tile (see Figure 12).

The next section focuses on using IBM Bluemix DevOps Services in order to change the code and modify the application while the application is running in production—with no downtime and with flawless integration, for providing continuous delivery.

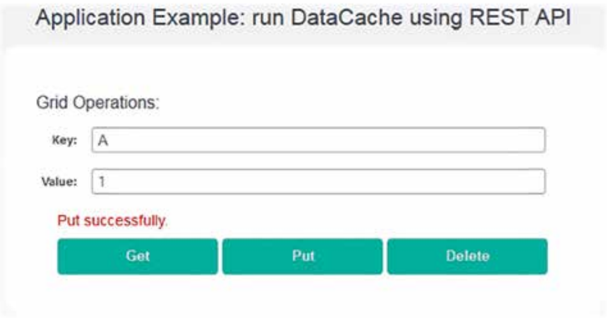


js-Demo

js-Demo.mybluemix.net

Running

Figure 12: **Application on the dashboard**



Application Example: run DataCache using REST API

Grid Operations:

Key: A

Value: 1

Put successfully

Figure 13: **Application on the Internet**

7.3. Modifying the application

You can change and modify the code by 3 different ways: Eclipse plugin for Bluemix, the Cloud Foundry command line interface and IBM Bluemix DevOps Services that I chose to proceed my tutorial with. During the tutorial, you will get to find more details about the other coding ways.

How to register for the IBM Bluemix DevOps Services:

1. Visit hub.jazz.net.

2. Click on REGISTER.

3. If you already have an IBM ID, jump to step 5.

4. Register for an IBM ID using your work email and follow the instructions.

5. Click on “Sign in to start using IBM Bluemix DevOps Services” (with your created IBM ID).

6. Click on “Create a Jazz ID” (you can use your IBM ID). You will receive a confirmation note for registering to jazz.net by e-mail.

7. Click on “Confirm & Link Accounts” to connect your IBM ID and Jazz ID.

a) Switch back to Bluemix and Click on Dashboard on top where you can click again on the application tile (see Figure 12).

b) An overview of the application will appear (see Figure 14).
   1. “Box 1” is a guide to the different methods in-order to change and modify your application that I have mentioned earlier.
   2. Click on “Box 2”, which features the URL of the application on the internet.
   3. “Box 3” ensures the integration of Bluemix with IBM DevOps, which is demonstrated later on.
   4. If you click on “Box 4”, it will show the details of the application from the number of instances to the RAM capacity.
   5. The instances to compensate with the workload can be increased and decreased with “Box 5”. This also can be done automatically using the “Auto-Scaling” service from the Bluemix Catalog.
   6. The memory quota is calibrated in “Box 6”.
   7. Allocated memory for applications is shown in “Box 7”.
   8. Start and restart the application is demonstrated in “Box 8”.
   9. “Box 9” is mentioned to “Add a new service or API” to your application.
   10. “Box 10” gives a cost estimate of the application.
   11. “Box 11” shows the unique credentials that allow to connect to this very service and interact with it through the application.

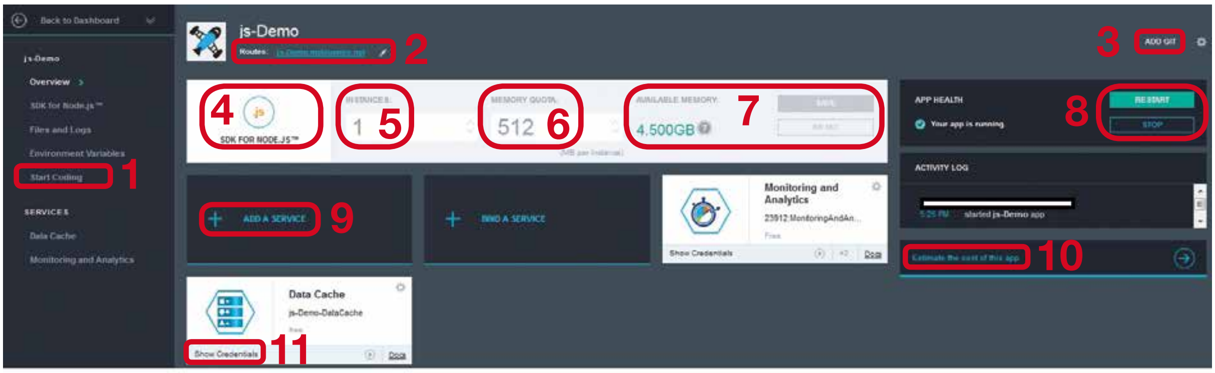


Figure 14: **Application Overview**

c) Click on ADD GIT in “Box 3” at the top right (see Figure 14).

d) Enter your IBM ID password, then click Sign In (Figure 15).

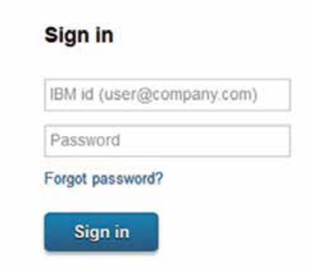


Figure 15: **IBM Bluemix DevOps Services sign-in**

Note: If you do not yet have access to IBM DevOps Services, you will be asked to create a “Jazz ID” for the first time and link it to your IBM ID. You can use your new Jazz ID to interact with the GIT repository allocated to your app.

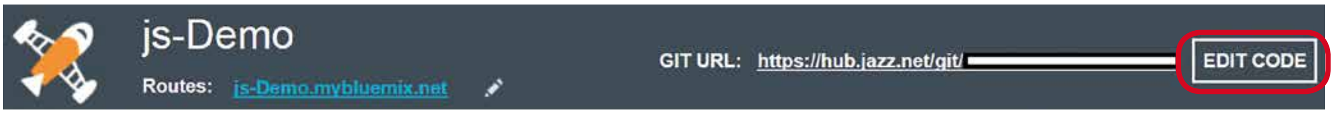
e) Check the box that reads “Populate the repository with the starter application package and enable build and deploy”, then click CONTINUE. Proceed till you are required to click CLOSE.

f) Done! A GIT repository has been created containing your code and an application deployed from Bluemix.

g) Click EDIT CODE (see Figure 16).

h) Now IBM Bluemix DevOps Services will open with the project deployed from Bluemix. It offers a vast array of functionality, but for now we will continue with our application modification.

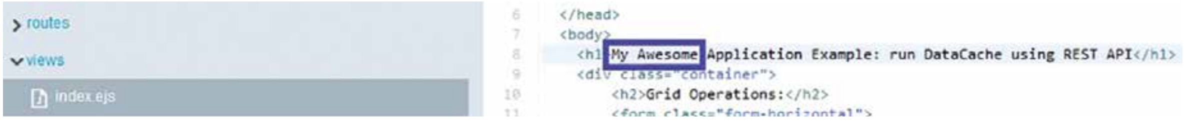
i) Navigate to the views folder on the left and click on the file index.ejs to modify the app. There is a piece of code on the right-hand side where the application title can be changed and add “My Awesome” at the beginning (see Figure 17).



Routes: js-Demo.mybluemix.net

GIT URL: https://hub.jazz.net/git/[redacted]

Figure 16: **Git URL**



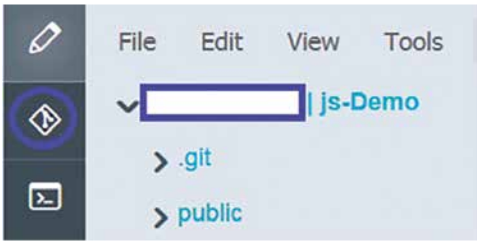
```
</head>
<body>
  <h1>My Awesome Application Example: run DataCache using REST API</h1>
  <div class="container">
    <h2>Grid Operations:</h2>
    <form class="form-horizontal">
```

Figure 17: **Editing code in IBM DevOps Services**

j) Save the change by pressing “CTRL+S” or click on “File” then “Save”.

k) Navigate to the tabs on the left-hand side and click on “Git Repository”, which has an icon indicated in the blue circle (see Figure 18).

l) The views/index.ejs file will appear in the Working Directory Changes section along with other files. Click on the blank box to enter a commit message and Select all files as indicated and click on COMMIT (Figure 19).

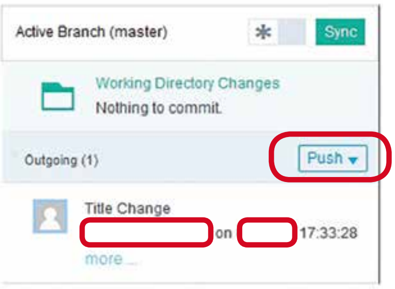


[redacted] | js-Demo
> .git
> public

Figure 18: **Git Status**

Note: Even though we have only modified one file, the other files only appear the first time we commit a change, as it relates to the Node.JS runtime.

m) The file is then transferred to the OUTGOING section. Select it and click PUSH (see Figure 20).



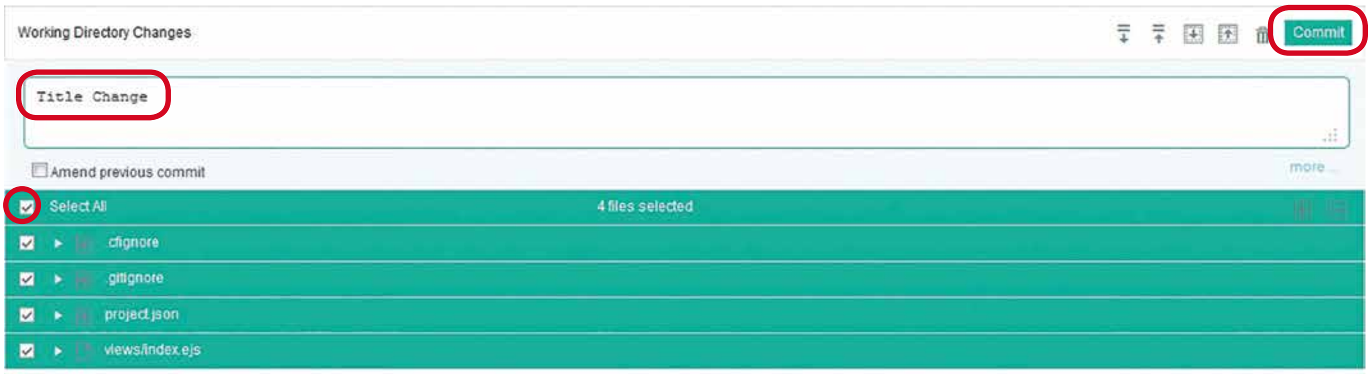
Active Branch (master)

Working Directory Changes
Nothing to commit.

Outgoing (1)

Title Change
[redacted] on [redacted] 17:33:28

Figure 20: **Push the update in IBM Bluemix DevOps Services**



Working Directory Changes

```
Title Change
```

□ Amend previous commit

☑ Select All 4 files selected

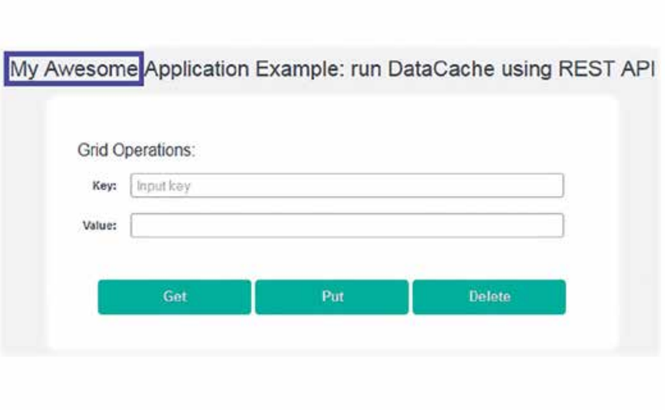
☑ ▸ .cfignore
☑ ▸ .gitignore
☑ ▸ project.json
☑ ▸ views/index.ejs

Figure 19: **Commit the updated file in IBM Bluemix DevOps Services**

n) Now go to the BUILD & DEPLOY page (top right corner) and click on it to check the status of our modification. The new deployment will top the list of changes on the BUILD & DEPLOY page and will automatically modify the existing application on Bluemix.

o) To check the modification on the application itself, refresh the browser where the application is running online.

p) The application update has been successfully and easily completed in a short space of time (see Figure 21). You will receive a confirmation note with your new Bluemix project stating the project url and your git repository url by e-mail. This is the end of the tutorial.



My Awesome Application Example: run DataCache using REST API

Grid Operations:

Key: Input key

Value:

Figure 21: **Updated version of the application**

8. Summary

Bluemix is an open-standard, cloud-based platform for building, managing, and running all types of applications (web, mobile, big data, new smart devices, etc.). It supports fit-for-purpose programming models and services, while supporting optimized, elastic workloads at the click of a button. A user-friendly interface ensures the manageability of the services and applications.

Moreover, Bluemix enables continuous delivery and availability by leveraging IBM Bluemix DevOps Services—an end-to-end solution in the cloud that delivers an open, integrated, rapid development experience that scales. It can change the way developers work by delivering results faster; in less than two minutes, developers can not only go from idea to search to coding via integrated developer tools, but you also have everything you need to write state-of-the-art mobile apps that are pre-integrated and preconfigured. This is all achieved with APIs and SDKs that can quickly and easily be incorporated into the latest technology.

With Bluemix, IBM is leveraging its middleware platforms and Cloud Foundry to enable enterprises to build systems based on interaction. Now companies can build next-generation applications that take advantage of existing investments in middleware, packaged applications, line-of-business applications, and business partner solutions (essentially systems of record) and integrate them with new cloud-centric services and applications (systems of engagement) deployed via cloud delivery models for workload-optimized deployment.

Why IBM Bluemix?

- Speed—time-to-value
  - Faster: Accelerate IT services' contribution to the business, from application development, to infrastructure deployment and monitoring.
  - Improved: Deploy high-value-generating applications quickly, instead of spending large amounts of time scripting and debugging.
- Agility—Greater operational efficiency
  - Adaptable: Scale environments elastically based on business demand.
  - Extensible: Transform legacy environments to a platform approach.
- Predictability—Reduced risk and cost
  - Improve application performance and availability.
  - De-risk and separate out the concerns of application development and operations.

For more information

Karim Abousedera
Cloud Specialist
Bluemix Subject-Matter Expert
IBM Deutschland GmbH
abousedera@de.ibm.com

IBM Deutschland GmbH
IBM-Allee 1
71139 Ehningen
ibm.com/de

IBM Österreich
Internationale Büromaschinen Gesellschaft m.b.H.
Obere Donaustraße 95
1020 Wien
ibm.com/at

IBM Switzerland Ltd.
Vulkanstrasse 106
8010 Zürich
ibm.com/ch

Printed in Germany
May 2015