

EXPERIENCING PLATFORMS :
Hands On with IBM Bluemix and IBM Watson

Ayush Dhoot

Developing the Industrial Internet of Things

Background:

Imagine not having to pay the cost and trying to figure out how to buy and manage software licenses, the underlying application infrastructure and middleware or the development tools and other resources for you to develop and manage the applications and services. Wouldn't it be just wonderful and simple? Just for such simplicity and efficiency, Platforms has been created.

Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. You purchase the resources you need from a cloud service provider on a pay-as-you-go basis and access them over a secure Internet connection. [1]

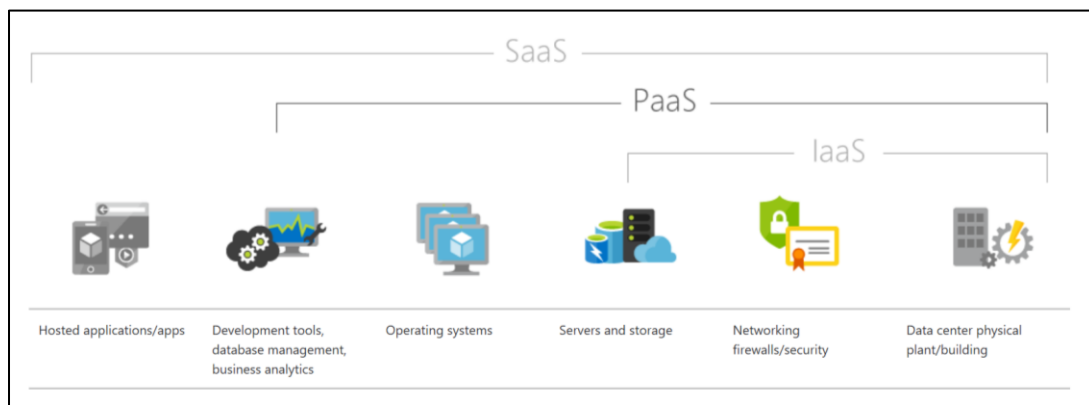


Figure 1 : Types of Cloud Computing Services

What I did :

For learning about the Platforms, I started right at the beginning. First step was to create an account with any of the platforms available. I chose to use IBM Bluemix as there was many free pre-loaded templates already present in their environment which would make the learning process easier. I created my account on Bluemix and logged in. Once I was in the system, there was many tutorials about the many different types of applications available. I started with the 'Introduce toolchains by using the "Develop a Cloud Foundry app" toolchain'. The introduction mentioned that this tutorial will help me create an open source toolchain which will act as a 'Hello World' app. The task was to create a toolchain, which is a set of tool integrations that support development, deployment, and operations tasks. I selected the 'Develop a Cloud Foundry app' toolchain from the Toolchain catalog. This led to a few reviews about the information and linking of my Git account to the toolchain and creating a new API key for the configuration. At the end of these several steps, I was able to create a toolchain. Then I just had to check the delivery pipeline to complete the process by building and deploying the toolchain. On successfully doing so, I was able to run the app which shows 'Hello World' on the output html page. I decided to then go ahead and try to modify the source code. For this, I used the Eclipse Orion Web IDE. I

changed the display of welcome message to 'Hello World Updated Again' in the index.html file. Then to incorporate the change, I had to use Git for pushing the changes to the master branch, as the delivery pipeline automatically runs the code from the master branch. All of these was done on the cloud itself and on the click of few buttons. Now I wanted to see what the delivery pipeline stages does so I added a new stage which would act as a test stage which would stop the app from deploying if the job fails. In this toolchain, I could deploy an app to the cloud and the changes made and pushed to the repo was also automatically deployed by the delivery pipeline. For simplicity, we can have the same toolchain for many apps where each one would have a repo and delivery pipeline of its own. [2]

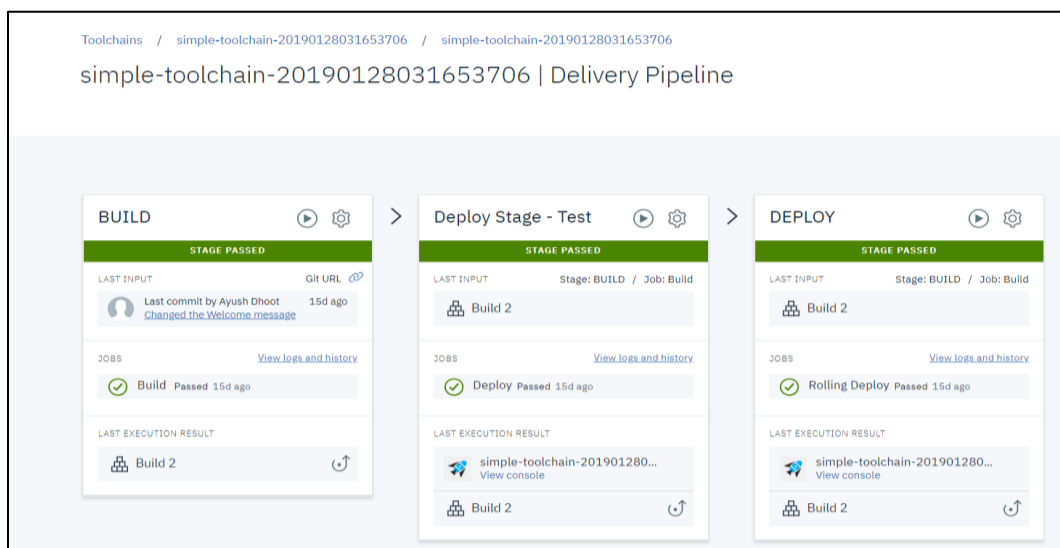


Figure 2 : Delivery Pipeline with addition of a test stage

The screenshot shows the Eclipse Orion Web IDE interface. The file explorer on the left displays the project structure, including 'Root (cloud.ibm.com)', 'simple-toolchain-20190128031653706', 'launchConfigurations', 'public', 'images', 'stylesheets', and 'index.html'. The main editor area shows the 'index.html' file being edited. The code content is as follows:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5   <title>Hello World</title>
6   <meta charset="utf-8">
7   <meta name="viewport" content="width=device-width, initial-scale=1">
8   <link rel="stylesheet" href="stylesheets/style.css">
9 </head>
10
11 <body>
12   <table>
13     <tr>
14       <td style="width:30%;>
15         
16       </td>
17     <td>
18       <h1>Hello world Updated Again!</h1>
19     </td>
20   </tr>
21 </table>
22 </body>
23
24 </html>
25
```

Figure 3 : Changes made in the Eclipse Orion Web IDE and pushed to the Git repo

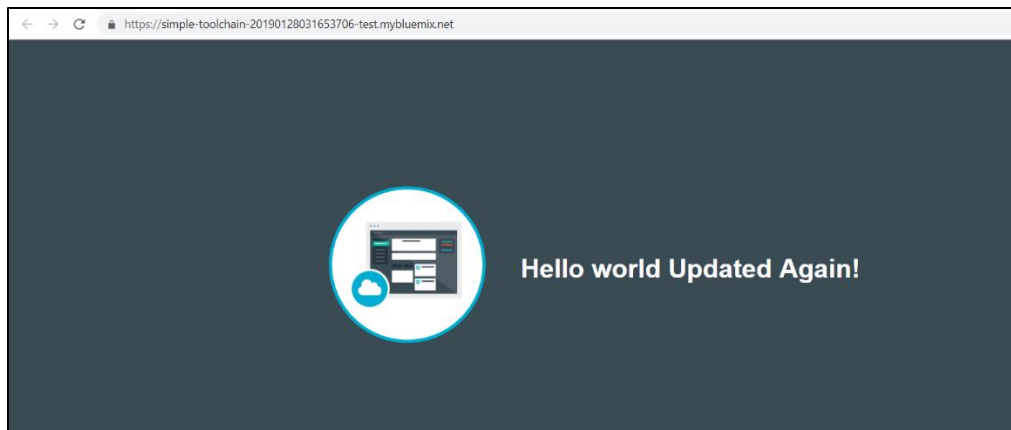


Figure 4 : Output of the Cloud Foundry App developed

I wanted to learn about the Watson, the AI system developed by IBM's DeepQA project, so I tried the Face Detection service offered by it. I began by building a node-RED Starter Boilerplate app from the Catalog tab of IBM cloud. After the login, I was taken to the Node-RED canvas where we could drag and drop nodes according to our application. For the facial recognition software, I needed to create a new service called Visual Recognition which will help in detecting faces. The nodes needed for the application was dragged and dropped onto the editor and then configured for our need. The flow was created by connecting the nodes such that it represents a webpage where the input will be an image URL and then the app was deployed. I changed the frontpage of the webpage to show my name and replaced the default images from the template. I did successfully build a face recognition app which analyzes the image and extract details about the image. The Face Detection service can identify multiple faces within the image and determine their gender and age with a confidence score and identify celebrities.

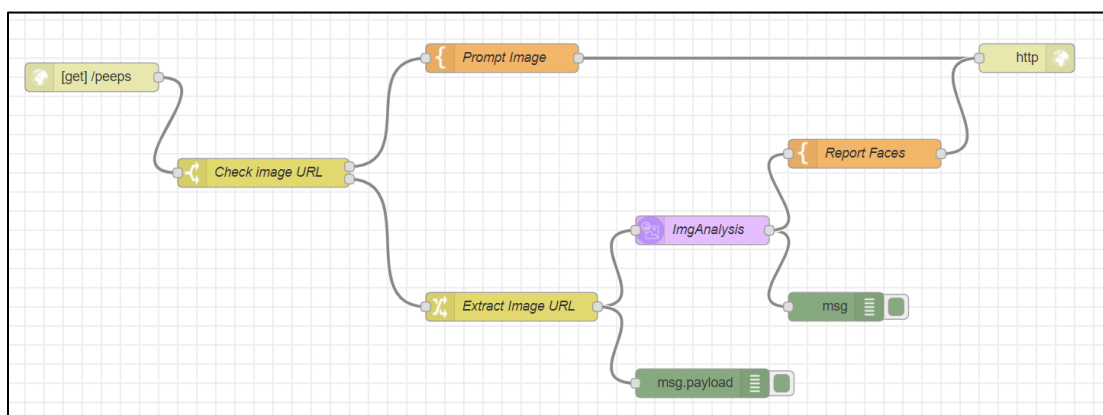


Figure 5 : Flow of the Image Detection service

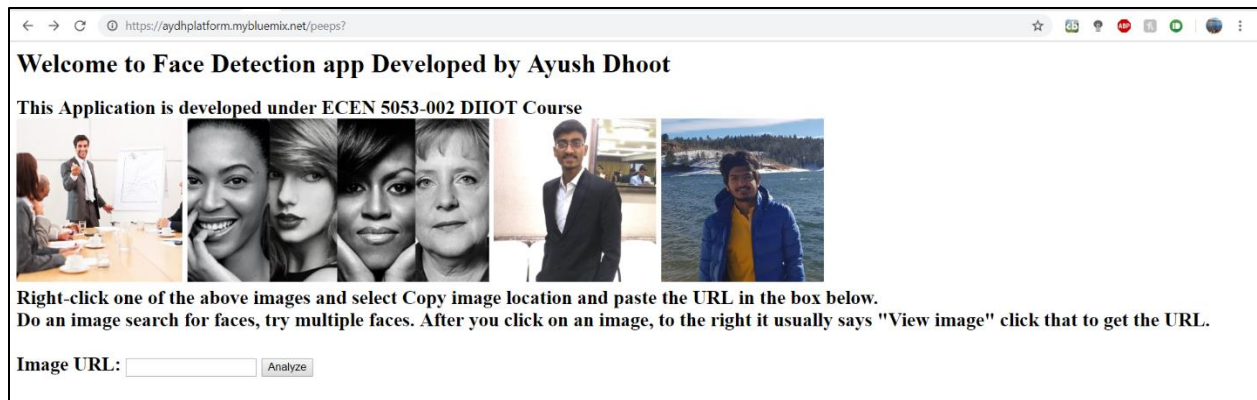


Figure 6 : Webpage of the Face Detection app

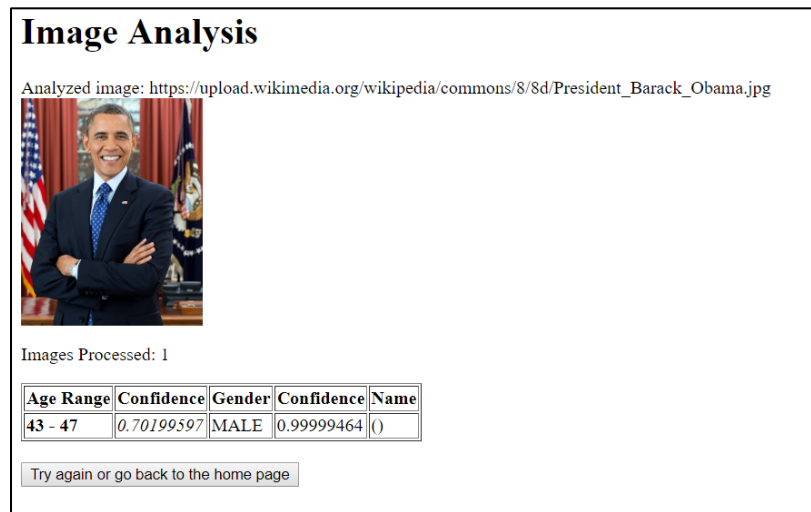


Figure 7 : Output from the analysis of the Image URL

To further increase the understanding of IBM Watson, I decided to do another task which was an app which determines real-time earthquakes. This app used the csv data from the resources of USGS and located the active earthquake on a global map. The flow for the same was constructed by dragging and dropping nodes and connecting them in a logical flow on the canvas. I even got to use one of the many open sourced nodes which is available in Node-RED to display the worldmap. The activation of the earthquake service can be done over a single click of the inject node and we can see the earthquakes on the world map. We can change the views and add more nodes(like heatmap) for better visual appearance.

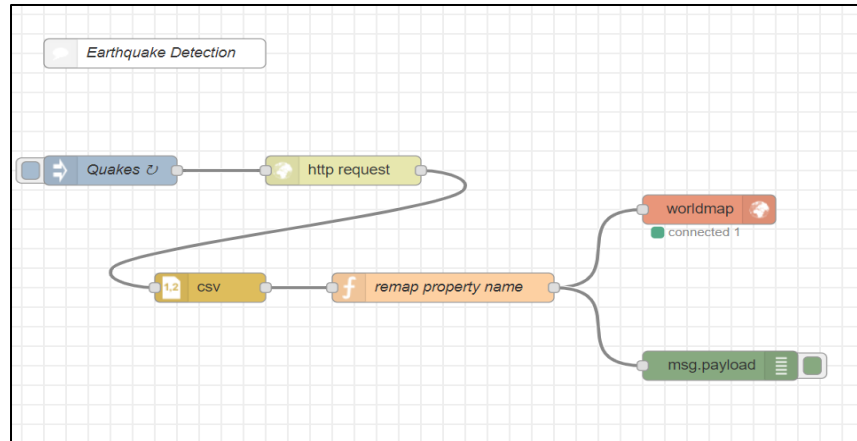


Figure 8 : Flow of the service which identifies earthquake on a worldmap

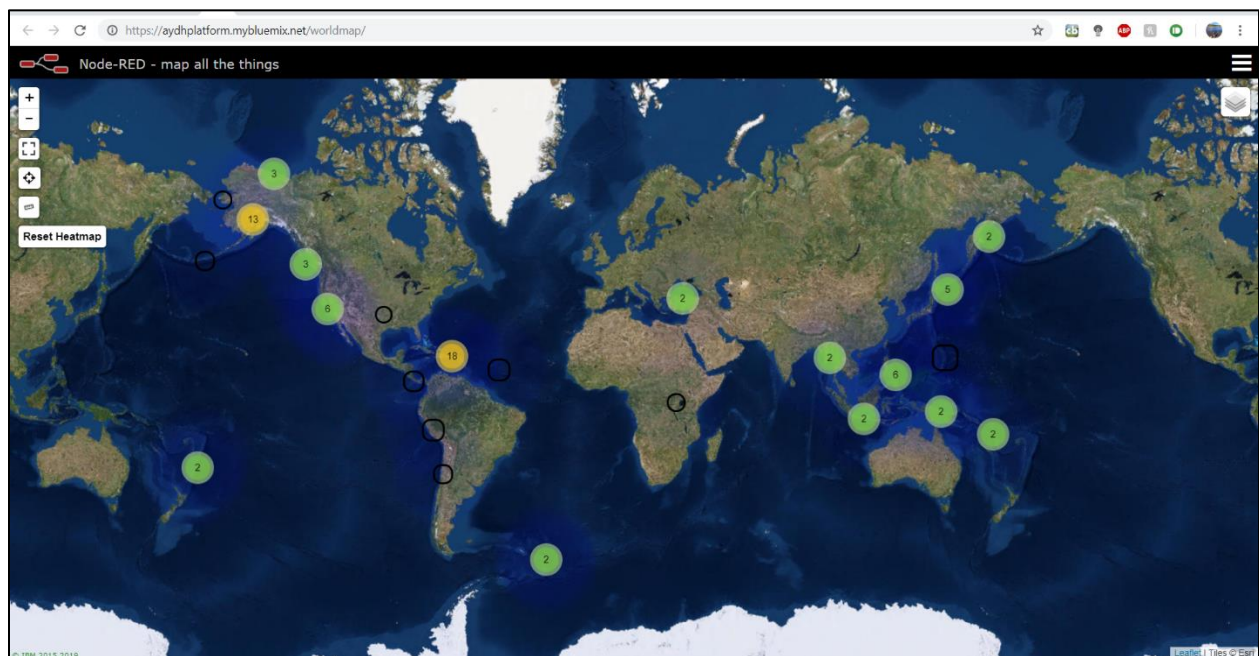


Figure 9 : Worldmap showing real-time earthquakes

What I Learned :

From the above experience of using IBM Bluemix environment to develop and deploy some apps, I have found that using platforms for such applications can be simple and efficient. We can develop framework where the users can build and run cloud-based services. They can drastically reduce the time needed to code from scratch as there are pre-loaded services present in the platform which can be used to develop the applications. Platform is an excellent tool for any kind of organizations or users. The ease of using the services and enhancing the outcome are few advantages of using a platform. In a platform, we can build, test, deploy, manage and update

within the same environment in the cloud. The usage of Platforms is not only cheaper as there are many different plans according to the need of the developer but also because the developer doesn't have to pay for the costly software and tools individually. As Platform are open-sourced and cloud-based, the services can be modified, deployed from any devices which is connected to the internet.

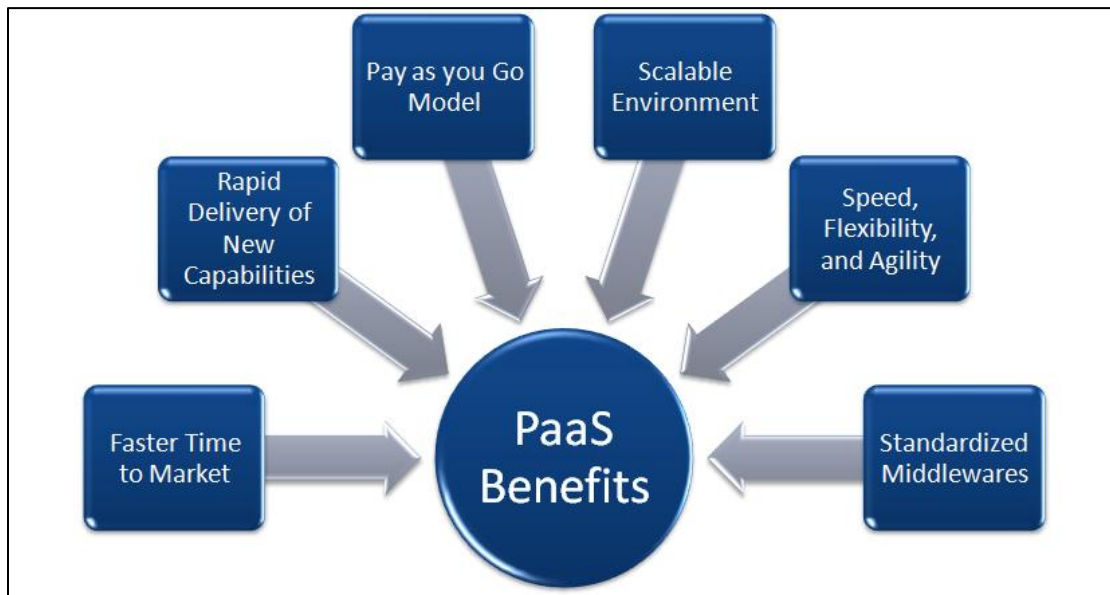


Figure 10 : Advantages of using Platforms

If I imagine myself owning a company, I think that using Platforms as a Service would be very beneficial for my company. It will help me increase productivity, reduce the expenditure by not having to pay for the software, not employing individuals for each task, have flexibility as the platform can be accessed from any of the cloud supporting devices. I can have a DevOps team which can take care of the whole process including building, debugging and deploying solely in the platform environment. There are many pre-coded services available, so the team will not have to start from the beginning but can use them as a head start which will reduce the coding time and have much faster deployment in the marketplace. As there are so many services available, the company could use them for more efficiency in the product development. Access to these services can result in much better functioning as they can be used for multiple purposes. We can use the Watson to have AI and Data related applications which can help the company to automate and learn more about the product and its market.

I found that IBM Bluemix is an excellent platform for developers as well as organization as it is quite simple and the tutorials are well defined. I was able to try out many services offered by IBM in its Bluemix and Watson environment. While exploring these services, I found out that using Platforms is very efficient in both cost and operation.

References:

- [1] <https://azure.microsoft.com/en-us/overview/what-is-paas/>
- [2] <https://www.ibm.com/cloud/garage/tutorials/introduce-develop-cloud-foundry-app-toolchain?task=5>
- [3] <https://www.znetlive.com/blog/wp-content/uploads/2016/06/Paas-Benefits.png>