DEVELOPMENT OF INTERNET OF THINGS ENABLED WEB APPLICATIONS USING RED HAT OPENSHIFT

Dr. Dharmeshkumar Bhalchandra Bhavsar
Associate Professor
Shri Chimanbhai Patel Institute of Computer Applications
Ahmedabad, Gujarat, India

Abstract

With the increase volume, velocity and variety of data from multiple channels, there has been huge demand for the high performance computing resources which can process the enormous as well as heterogeneous big data. It is not always possible to purchase the costly computing resources including high performance multi core processors having supercomputing powers, huge memory devices and related technologies to process, visualize and predict on the datasets related to live streaming and real time supercomputing applications. To cope up and work with such technologies, the cloud services are used from where the computing resources can be hired on demand and billing can be done as per the usage. Using this approach, there is billing of computing resources on demand as per the usage and processing time. The paper presents the IoT based environment of Red Hat OpenShift with the development scenarios.

Keywords: Internet of Things, IoT, Cloud based IoT, CoT

Introduction

A number of cloud services providers are available in the global market with different IoT and cloud delivery models including Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). Now days, there are some new keywords in cloud delivery including Network as a Service (NaaS), Database as a Service (DBaaS), Testing as a Service (TaaS) and many others depending upon the type of implementation [1, 2]. Each of these cloud delivery approaches are having different cloud resources which are used for different applications [3, 4].

Following is the table presenting key cloud service providers for different domains and applications

Table 1: Cloud and IoT based Services

IoT and Cloud Solutions	**Service Providers**
IaaS	• Amazon Web Services • Microsoft Azure • Google Compute Engine • Rackspace Open Cloud • IBM SmartCloud • HP Enterprise Converged Infrastructure
PaaS	• Engine Yard • RedHat OpenShift • Google App Engine • Heroku • Caspio • Microsoft Azure Cloud Services • AppFog • Amazon Web Services
SaaS	• Google Docs • Zoho • Office Suite Providers

	• Help Desk Providers • SalesForce • Citrix GoToMeeting • Cisco WebEx

Features and Images in Red Hat OpenShift for IoT and Web Apps

RedHat OpenShift is one of the leading cloud service providers under the paradigm of PaaS (Platform as a Service). OpenShift provides the multiple platforms to the cloud users with flexibility to develop, deploy and execute the applications on cloud [5, 6].

OpenShift is having high performance data centers with the enormous processing power to work with different programming languages which includes the following

- Java
- PHP
- Ruby
- Python
- Node.js
- Perl
- Jenkins Server
- Ghost
- Go
- Mean
- Cartridge Development Kit
- Sinatra

and many others

A beginner can use the Free Tier of RedHat OpenShift for development, deployment and execution of new cloud apps on the online platform provided by OpenShift. Any of the mentioned programming language can be used for the development of apps with the real time implementation [7, 8].

Developing IoT based Web Applications

OpenShift provides multiple options for programming languages to the cloud user for development of the apps. With each programming language, OpenShift delivers multiple versions so that the compatibility issues can be avoided in later stage.

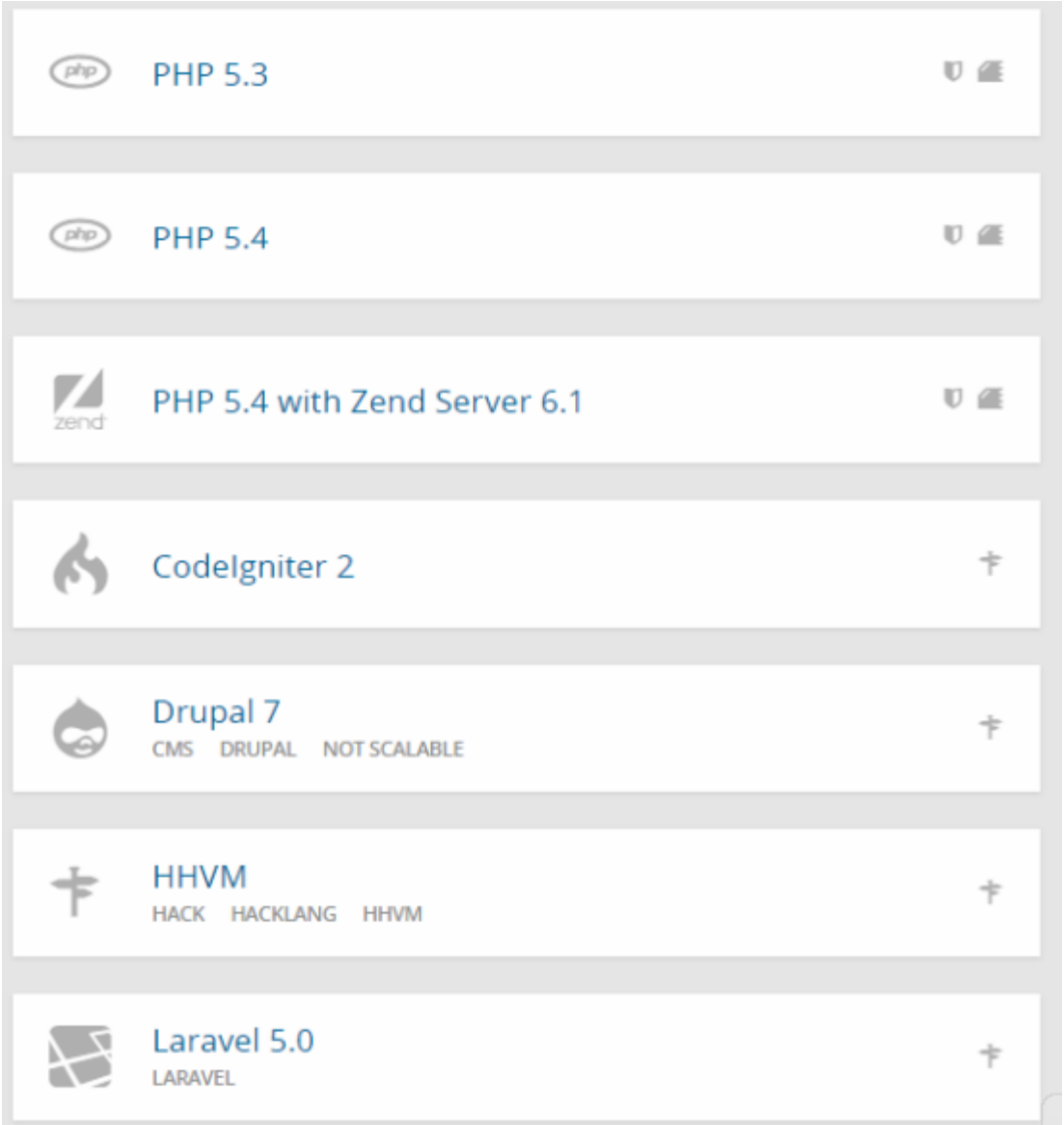

Figure 1: Selection of the Programming Language on OpenShift

The cloud applications can be uploaded using mapping with GIT via local command prompt (Windows CMD or Linux Terminal). OpenShift specifies the commands which should be executed on local command prompt or Linux Shell.

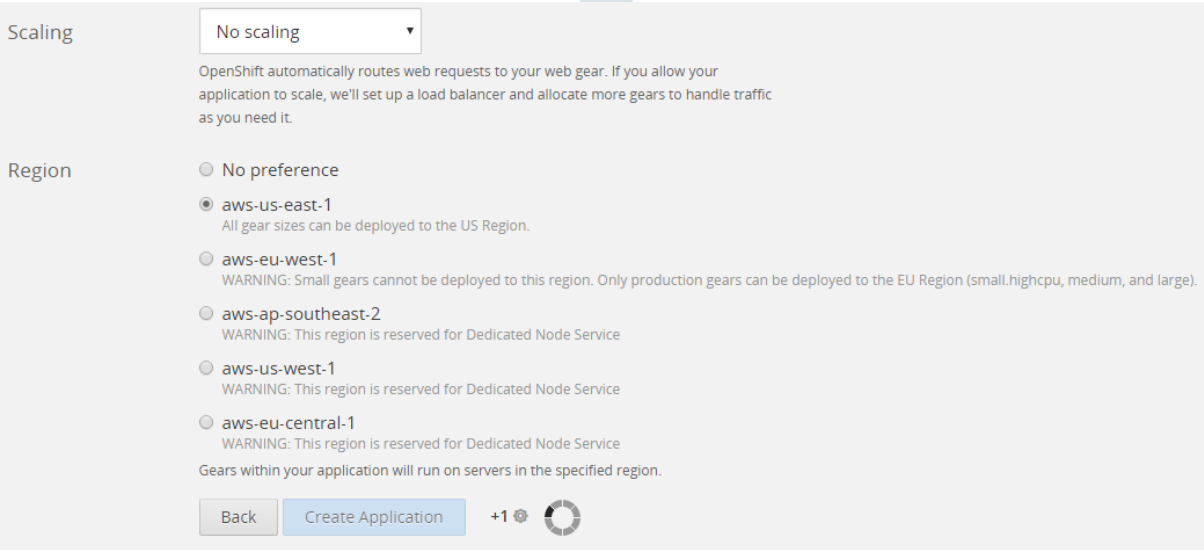

Figure 2: Selection of the region for deployment of cloud application

```
RSA key fingerprint is cf:ee:77:cb:0e:fc:02:d7:72:7e:ae:80:c0:90:88:a7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'php-magmaambala.rhcloud.com,52.21.212.8' (RSA) to th
e list of known hosts.
remote: Counting objects: 19, done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 19 (delta 0), reused 19 (delta 0)
Receiving objects: 100% (19/19), 18.25 KiB | 0 bytes/s, done.
Checking connectivity... done.
```

Figure 3: Mapping of Command Prompt with GIT

PHP based Machine Learning on OpenShift PaaS

URL : https://php-ml.readthedocs.io/en/latest/

Machine Learning is one the powerful approach which makes use of soft computing and metaheuristic approaches for effectual predictive mining from even a huge dataset. Machine learning is traditionally used for frauds detection, market analytics, email spam filtering, malware analysis, fingerprint evaluations, face detection and many other applications. In machine learning, the algorithms are implemented in such a way in which better classification and predictions can be done that is similar to the intelligence in human brain. In traditional implementations, the artificial neural networks are used with machine leaning for solving the complex classification problems.

A number of libraries and frameworks are available under Free and Open Source Software (FOSS) distribution for machine learning. Following are few libraries which are free and open source and can be integrated for research and development

- PHP-ML
- Apache Mahout
- Shogun
- Apache Singa
- Apache Spark MLib
- TensorFlow
- Oryx2
- Accord.NET
- Amazon Machine Learning
- Scikit-Learn
- H_2O
- ConvNetJS

PHP-ML is one of the powerful machine learning libraries that is used for research and development in the domain of machine learning for different applications. PHP-ML integrates assorted algorithms in form of classes and methods for high performance computing with the analytics from real time datasets. PHP-ML is having a rich set of algorithms implemented in PHP Scripts and these can be easily integrated on real cloud of OpenShift by uploading the code and mapping with GIT.

Key Features and Algorithms in PHP-ML

Association Rule Learning

- Apriori

Classification

- k-Nearest Neighbors
- Naive Bayes
- SVC

Clustering

- k-Means
- DBSCAN

Cross Validation

- Random Split
- Stratified Random Split

Feature Extraction

- Token Count Vectorizer
- Tf-idf Transformer

Metric
- Accuracy
- Confusion Matrix
- Classification Report

Models management
- Persistency

Math
- Distance
- Matrix
- Set
- Statistic

Neural Network
- Multilayer Perceptron Classifier

Preprocessing
- Normalization
- Imputation missing values

Regression
- Least Squares
- SVR

Workflow
- Pipeline

Datasets
- Array
- CSV
- Files
- Datasets for Research : Iris, Wine and Glass

KNearestNeighbors Classifier in PHP-ML

KNearestNeighbors implements the k-nearest neighbors (k-NN) algorithm for solving the classification problems for specific set of data items. Following is the example in which the inputs with their corresponding targets are specified in terms of classes '0' or '1'. If these values are carefully analyzed, the corresponding classes can be mapped. In the dataset of [2, 5], [3, 6], [4, 7] the values in each set is in increasing order and there is difference of +3 and assigned the class '0'. Similarly, in [4, 2], [5, 3], [7, 5], the values in each set is decreasing and assigned class '1'. Now, this data can be trained using k-NN with the implementation of train() function.

```
$input = [[2, 5], [3, 6], [4, 7], [4, 2], [5, 3], [7, 5]];
$target = ['0', '0', '0', '1', '1', '1'];
$classifier = new KNearestNeighbors();
$classifier->train($input, $target);
```

For prediction of new input data, the predict() function is implemented. As in the following example predict([5, 7]) is passed as input, the output will be return as class '0' because the values in [5, 7] are in increasing and almost same behavior as class '0'. The exact difference of +3 is not mandatory because machine learning approaches make use of results with higher degree of approximation, probability and optimization.

Conclusion

As there are many applications where classification and predictive mining can be used, the free and open source libraries can be integrated on real clouds of Red Hat OpenShift, IBM Bluemix, Amazon, Google Apps Engine and many others depending upon the algorithms to be used. The aspects and logs associated with performance, complexity, security and integrity can be analyzed with the implementation of algorithms on real time clouds.

References

[1] Doukas, C., & Maglogiannis, I. (2012, July). Bringing IoT and cloud computing towards pervasive healthcare. In Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on (pp. 922-926). IEEE.
[2] Wang, C., Bi, Z., & Da Xu, L. (2014). IoT and cloud computing in automation of assembly modeling systems. IEEE Transactions on Industrial Informatics, 10(2), 1426-1434.
[3] Biswas, A. R., & Giaffreda, R. (2014, March). IoT and cloud convergence: Opportunities and challenges. In 2014 IEEE World Forum on Internet of Things (WF-IoT) (pp. 375-376). IEEE.
[4] Stergiou, C., Psannis, K. E., Kim, B. G., & Gupta, B. (2018). Secure integration of IoT and cloud computing. Future Generation Computer Systems, 78, 964-975.
[5] Antonopoulos, N., & Gillam, L. (2010). Cloud computing. London: Springer.

[6] Walraven, S., Monheim, T., Truyen, E., & Joosen, W. (2012, December). Towards performance isolation in multi-tenant saas applications. In Proceedings of the 7th Workshop on Middleware for Next Generation Internet Computing (p. 6). ACM.

[7] Bernstein, D. (2014). Cloud foundry aims to become the OpenStack of PaaS. IEEE Cloud Computing, 1(2), 57-60.

[8] Rafique, A., Walraven, S., Lagaisse, B., Desair, T., & Joosen, W. (2014, April). Towards portability and interoperability support in middleware for hybrid clouds. In IEEE Conference on Computer Communications Workshops (INFOCOM), 2014 (pp. 7-12). IEEE.