**Decision Engine Platform**

### BITS ZG628T: Dissertation

by

Pramod Kumar N

2014HT13292

# Dissertation work carried out at

## Coextrix Technologies Pvt. Ltd., Bengaluru

****

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE**

**PILANI (RAJASTHAN)**

November 2016

**Decision Engine Platform**

### BITS ZG628T: Dissertation

by

Pramod Kumar N

2014HT13292

# Dissertation work carried out at

## Coextrix Technologies Pvt. Ltd., Bengaluru

Submitted in partial fulfillment of M.Tech. Software Systems degree programme

Under the Supervision of

Mr.Ramesh Krishnamoorthy, CEO,

Coextrix Technologies Pvt. Ltd., Bengaluru

****

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE**

**PILANI (RAJASTHAN)**

November, 2016

#### 

#### CERTIFICATE

This is to certify that the Dissertation entitled Decision Engine Platform and submitted by Pramod Kumar N having ID-No. 2014HT13292 for the partial fulfillment of the requirements of M.Tech. Software Systems degree of BITS, embodies the bonafide work done by him under my supervision.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of the Supervisor

Place : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name, Designation & Organization &Location

iii

**Birla Institute of Technology & Science, Pilani**

**Work-Integrated Learning Programmes Division**

**First Semester 2016-2017**

**BITS ZG628T: Dissertation**

**ABSTRACT**

**BITS ID No.** **: 2014HT13292**

**NAME OF THE STUDENT** **: Pramod Kumar N**

**EMAIL ADDRESS** **: pramod974@gmail.com**

**STUDENT’S EMPLOYING** **: Coextrix Technologies, Bengaluru**

**ORGANIZATION & LOCATION**

**SUPERVISOR’S NAME** **: Ramesh Krishnamoorthy**

**SUPERVISOR’S EMPLOYING** **: Coextrix Technologies, Bengaluru**

**ORGANIZATION & LOCATION**

**SUPERVISOR’S EMAIL ADDRESS : ramesh.krishnamoorthy@coextrix.com**

**DISSERTATION TITLE** **: Decision Engine Platform**

iv

**Abstract**

In modern day applications that provide more visibility into data irrespective of domain, require collecting data from several different sources. Collected data needs to be consolidated, normalized and then presented through an application. There is a lack of correctness in the data aggregated owing to the nature of issues with the sources of data itself. Since the original data sources are doing nothing to clean up the data they provide or generally there is no standardization maintained across domains in context. Several applications are designed to clean up the data coming through various sources and allow customers to perform complex analytics on top of the data. Transforming and reconciliation process is extensively complex, but is required to create higher accuracy in the data. The transformation and reconciliation process comprises of multiple rules, interactions with multiple components in order to achieve the accuracy. In popular existing open source technologies the way the rules are written and configured is stereotype and hard to maintain owing to the quantity of rules modelled and its ability to scale in future.

There are no matured open source decision engines built using python which satisfies applications current needs, where there is an end to end feature to model and maintain rules which could interact with multiple sources and components. Accurate data has been always sought for and is the next gen thing to get custom insights irrespective of domain. The underlying need to build a configurable and maintainable platform is highly compelling, where a Decision engine facilitates connecting to various sources and components in order to write domain specific configurable rules so as to achieve valuable decision in limited turnaround time. The Goal is to build, deploy and scale the existing stereotype rules/models to a more maintainable and configurable Decision engine and also provide an easy interface and framework to model rules on the fly build, test and deploy it without any manual intervention.

Broad Academic Area of Work: **Software Architecture**

Key words: **Rules, Engine, Decision, Python, Platform, Framework, Configurable**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of the Student**

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date:**

**Place:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of the Supervisor**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Date:**

**Place:**

v

**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of dissertation would be incomplete without mentioning the people who made it possible, whose constant guidance and encouragement crowns all the efforts with success.

I express sincere gratitude to Mr.Ramesh Krishnamoorthy, CEO, Coextrix for being kind enough to provide me an opportunity to work on a project in this organization and also being my mentor.

I am thankful to Mr. Ganesh Guruswamy, CTO, Coextrix for providing valuable ideas , suggestions, facilities, co-operation and his encouragement at all moments of my approach towards tackling the problem.

Last but not the least, I would like to express my love and gratitude to my beloved family and friends, for their understanding & motivation, through the duration of this project.

vi

**Table of Contents**

Certificate………………………………………………………………………………………………………………………………iii

Abstract…………………………………………………………………………………………………………………………………iv

Acknowledgment………………………………………………………………………………………………………………….vi

Table of Figures ix

Chapter 1: Introduction 1

1.1 IBM Notes and Domino 2

1.2 What is Cloud Computing? 2

1.3 3-4-5 Rule of Cloud Computing 4

Chapter 2: PaaS and Cloud Application Development 8

2.1 Different Cloud Service Provider 8

2.2 Windows Azure 9

2.2.1 Web Role v/s Worker Role 10

2.2.2 Azure Fabric Controller 11

2.2.3 Azure Cloud Storage Services 11

2.3 Google App Engine 12

2.3.1 Developing and deploying app on Google App Engine 12

2.4 IBM Bluemix 12

2.4.1 Bluemix Architecture 13

2.4.2 Deployment options based on Access Models 14

2.4.3 Deployment options based on Compute Infrastructure 15

2.5 Cloud Native Applications 16

2.5.1 How cloud applications are different? 16

2.5.2 Characteristics of Native Cloud Applications 16

Chapter 3: Project Overview 18

3.1 Which PaaS to use? 18

3.1.1 Application Platform Comparison 18

3.2 Why IBM Bluemix? 19

3.3 How IBM Bluemix works? 20

Chapter 4: Project Implementation 21

4.1 Application Development in IBM Domino Designer 22

4.1.1 Why XPages? 23

4.1.2 XPages Sample Code 24

4.2 Extension Library 26

Chapter 5: Domino XPages in Bluemix 27

5.1 XPages Runtime 27

5.2 Domino Data Service 27

5.3 XPages Boilerplates 28

Chapter 6: Testing and Integration 29

6.1 Creating New Applications in Bluemix 29

6.1.1 IBM Bluemix Manifest Code 30

6.2 Pushing Existing Applications to Bluemix 31

6.3 Scaling XPages Application in IBM Bluemix 34

Summary 35

Conclusion and Recommendations 36

Directions for future work 37

References 39

Checklist for the items in the report 40

**Table of Figures**

Figure 1: Overview of cloud computing 2

Figure 2: Convergence of technology fields making cloud computing 3

Figure 3: Cloud service models 4

Figure 4: SPI Model 5

Figure 5: Public Cloud 6

Figure 6: Hybrid Cloud 7

Figure 7: Different PaaS providers 9

Figure 8: Quick view of Azure Web Portal 9

Figure 9: Compute service with two Web roles and two Worker roles 10

Figure 10: Schematic diagram of Azure platform services 11

Figure 11: Bluemix Catalogue 13

Figure 12: Bluemix Architecture 14

Figure 13: Bluemix deployment options 15

Figure 14: IBM Bluemix - XPage Runtime 20

Figure 15: Deploying application on Bluemix 20

Figure 16: Invoking a Bluemix application 21

Figure 17: Basic IBM Notes Form Design 22

Figure 18: XPage Design and Source 23

Figure 19: openntf XPages Extension Library web page 26

Figure 20: XPages Runtime 27

Figure 21: Domino Data Service 27

Figure 22: XPages Boilerplates 28

Figure 23: Built-In Bluemix to Designer 29

Figure 24: Import IBM Bluemix Starter Code 30

Figure 25: NoSQL Database Service Dashboard 31

Figure 26: NoSQL Data Deployment 33

Figure 27: Bound the Services to IBM Bluemix 33

Figure 28: Sample Application on IBM Bluemix 34

Figure 29: Scaling the XPages Application 34

Figure 30: Migration Comparison based on different platforms 36

Ix

**GLOSSARY**

REST – Representational State Transfer

API – Application Program Interface

CSV – Comma Separated File

MSEXCEL – Microsoft Excel

**Chapter 1: Introduction**

**1.1 Data Sources**

Data has been most sought for in the modern era to gain deeper and meaningful insights into ones business. The insight from data is required irrespective of which domain one is working on, in order to gain such insight data has to be curated from multiple sources. The data source [1] is can be typically a connection set up to a carious computer databases running as a server or it could be as simple as a file (CSV, text, MSEXCEL, etc.) or it could just be a stream of data coming in via API or live feed.

**1.2 Problems and Challenges with Data**

The volume, correctness and consistency pose a major challenge with the data collected from various sources to be processed before they can be analysed. Some businesses require domain specific incorporations as well to meet the data quality issues. Some of the common data related problems and challenges are as follows.

1. Poor data quality such as noisy data, dirty data, missing values, inexact or incorrect values, inadequate data size and poor representation in data.

2. Integrating conflicting or redundant data from different sources and forms: multimedia files (audio, video and images), geo data, text, social, numeric, etc…

3. Proliferation of security and privacy concerns by individuals, organisations and governments.

4. Unavailability of data or difficult access to data.

5. Efficiency and scalability issues to effectively extract the information from huge amount of data in databases.

6. Dealing with huge datasets that require distributed approaches, dealing with non-static, unbalanced and cost-sensitive data.

7. Constantly updating model to handle data velocity or new incoming data.

8. High cost involved in buying and maintaining powerful softwares, servers and storage hardwares that handle large amounts of data.

9. Processing of large, complex and unstructured data into a structured format.

10. Sheer quantity of output from the data curated.

**Page 1 of 40**

**1.3 Data Transformation and Reconciliation**

Owing to multi natured problems with collected data, it has to be transformed and reconciled. Typically along with all standard process available for transformation and reconciliation, business use domain specific rules and knowledge to transform and reconcile the data so they are analysis ready.

Data transformation is the process of converting data or information from one format to another, typically from the format of a source system into the required format of a new destination system. The general process involves converting documents, but data conversions sometimes involve the conversion of a program from one computer language to another to enable the program to run on a different platform. The usual reason for this data migration is the adoption of a new system that's totally different from the previous one.[4]

In real practice, data transformation involves the use of a special program that's able to read the data’s original base language, determine the language into which the data that must be translated for it to be usable by the new program or system, and then proceeds to transform that data.

Data Transformation involves two key phases:

Data Mapping: The assignment of elements from the source base or system toward the destination to capture all transformations that occur. This is made more complicated when there are complex transformations like many-to-one or one-to-many rules for transformation.

Code Generation: The creation of the actual transformation program. The resulting data map specification is used to create an executable program to run on computer systems.

**Industrial process data validation and reconciliation**, or more briefly, **data validation and reconciliation (DVR)**, is a technology that uses process information and mathematical methods in order to automatically correct measurements in industrial processes. The use of DVR allows for extracting accurate and reliable information about the state of industry processes from raw measurement data and produces a single consistent set of data representing the most likely process operation.[5]

**Page 2 of 40**

**1.4 Software Arcitecture**

**Figure 1: Overview of cloud computing**

As per Wikipedia [3] definition, Cloud Computing is –

“*a model for enabling ubiquitous, convenient, on-demand access to a shared pool of* *configurable computing resources.*”

In the simplest terms, cloud computing means storing and accessing data and programs

**Page 2 of 40**

**Chapter 2: PaaS and Cloud Application Development**

Platform as a Service (PaaS) model provides a platform on which users can directly develop and deploy their applications without worrying about the complexity of setting up the hardware or system software. Should application developers be burdened by tasks of ensuring that a specific server is up and running? Should they worry about the disk space? Should they worry about which OS their app should support? The focus should be on solving the big problems. The compute infrastructure, platform, libraries and application deployment should all be automated and abstracted. This is where Cloud Computing i.e. PaaS plays a major role.

The PaaS model provides the tools within an environment needed to create applications that can run in Software as a Service model. PaaS is application middleware offered as a service to developers, integrators, and architects. Development and Operation teams use PaaS to design, build, and deliver customized applications or information services. Instead of relying on standardized SaaS, teams using PaaS have more control over solution architecture, quality of service, user experience, data models, identity, integration, and business logic. In PaaS you are given a toolkit to work with, a virtual machine to run your software on and it is up to you to design the software and its user-facing interface in a way that is appropriate to your needs. So PaaS systems range from full-blown developer platforms like Windows Azure to systems like Drupal, Squarespace, Wolf, and others where the tools are modules that are very well developed and require almost no coding. PaaS solution will ensure the availability of the application despite downtime of the underlying virtual machine by automatically creating a new instance of the application on a new virtual machine when the machine goes down. PaaS systems can be used to host a variety of cloud services –

Online portal-based applications like Facebook that need to scale to thousands of Users

Startup who wants to host their new application in a Software-as-a-Service model Can also be used for massively parallel computations

Enterprises can deploy their Line-of-Business applications in the cloud, taking advantage of the scale and availability while still maintaining security and privacy of data

1. **Different Cloud Service Provider**

In this section, we will have a quick look on different cloud service provider especially for PaaS model. There are numerous PaaS service provider in the current market. Below picture captures few of them –

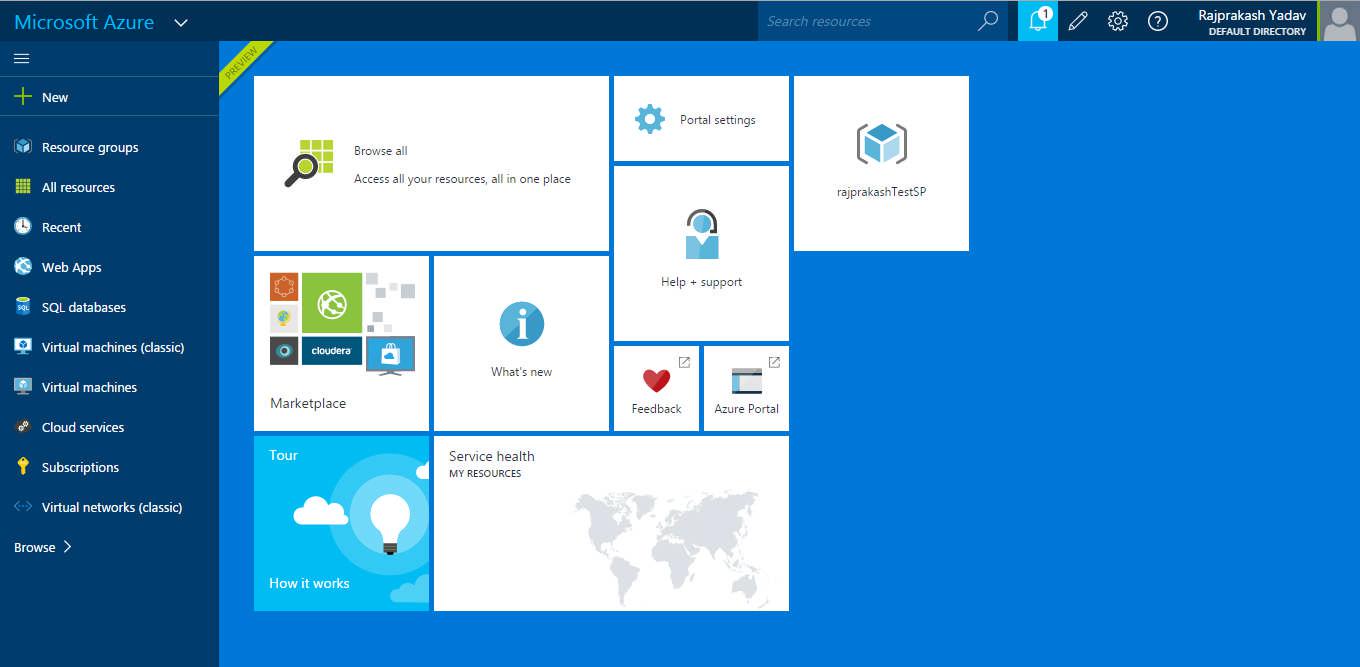
**Page 8 of 40**



**Figure 7: Different PaaS providers**

**2.2 Windows Azure**

The Azure Services Platform [6] is a popular application platform for the cloud that allows Windows applications and web-services to be hosted and run in Microsoft datacenters. It is cloud deployment platform for applications developed for Windows using .NET.



**Figure 8: Quick view of Azure Web Portal**

**Page 9 of 40**

**References**

1. Data Source Wiki, https://en.wikipedia.org/wiki/Data\_source
2. Computer File Wiki, https://en.wikipedia.org/wiki/Computer\_file
3. Data Stream Wiki, https://en.wikipedia.org/wiki/Data\_stream
4. Definition, https://www.techopedia.com
5. Data Reconciliation-https://en.wikipedia.org/wiki/Data\_validation\_and\_reconciliation
6. Google App Engine: Platform as a Service, https://cloud.google.com/appengine/docs
7. IBM Bluemix Docs, https://www.ng.bluemix.net/docs/
8. Cloud Foundry, https://www.cloudfoundry.org/
9. Cloud Native Applications on Bluemix - http://www.redbooks.ibm.com/redpieces/abstracts/sg248275.html?Open
10. Building Cloud Native Applications, Ryan Baxter, http://ryanjbaxter.com/2015/07/13/building-cloud-native-applications/
11. IBM, Microservices from Theory to Practice, IBM Redbooks, August, 2015
12. Heroku, Cloud Application Platform, https://www.heroku.com/
13. The Twelve-Factor App, http://12factor.net/
14. Migrating IBM Lotus Notes and Domino to the Cloud: Selecting Microsoft versus Google, December, 2010
15. IBM ConnectED 2015, Domino XPages for IBM Bluemix, http://www.ibm.com/developerworks/cloud/library/cl-bluemixfoundry/
16. Responsive Design, http://whatis.techtarget.com/definition/responsive-design
17. OpenNTF, http://www.openntf.org/
18. IBM Bluemix, The Digital Innovation Platform, https://www.ng.bluemix.net
19. 5 Things to Know about Why You Should Register for Bluemix, https://www.ibm.com/developerworks/community/blogs/5things/entry/5\_things\_to\_know\_ about\_why\_you\_should\_register\_for\_bluemix?lang=en

**Page 38 of 40**

**Checklist for the items in the report**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. | **Is the final report neatly formatted with all the elements required** | | Yes |  |
|  | **for a technical Report?** | |  |  |
|  |  | |  |  |
| 2. | Is the Cover page in proper format as given in Annexure A? | | Yes |  |
|  |  | |  |  |
| 3. | Is the Title page (Inner cover page) in proper format? | | Yes |  |
|  |  |  |  |  |
| 4. | (a) | Is the Certificate from the Supervisor in proper format? | Yes |  |
|  | (b) | Has it been signed by the Supervisor? | Yes |  |
|  |  | |  |  |
| 5. | Is the Abstract included in the report properly written within one page? | | Yes |  |
|  | Have the technical keywords been specified properly? | | Yes |  |
|  |  | |  |  |
| 6. | Is the title of your report appropriate? **The title should be adequately** | | Yes |  |
|  | **descriptive, precise and must reflect scope of the actual work done.** | |  |  |
|  | Uncommon abbreviations / Acronyms should not be used in the title | |  |  |
|  |  | |  |  |
| 7. | Have you included the List of abbreviations / Acronyms? | | Yes |  |
|  |  | |  |  |
| 8. | Does the Report contain a summary of the literature survey? | | Yes |  |
|  |  | |  |  |
| 9. | Does the Table of Contents include page numbers? | | Yes |  |
|  | (i). | Are the Pages numbered properly? (Ch. 1 should start on Page # 1) | Yes |  |
|  | (ii). | Are the Figures numbered properly? (Figure Numbers and Figure | Yes |  |
|  |  | Titles should be at the bottom of the figures) |  |  |
|  | (iii). Are the Tables numbered properly? (Table Numbers and Table Titles | | Yes |  |
|  |  | should be at the top of the tables) |  |
|  | (iv). | Are the Captions for the Figures and Tables proper? | Yes |  |
|  | (v). | Are the Appendices numbered properly? Are their titles appropriate | Yes |  |
|  |  | |  |  |
| 10. | Is the conclusion of the Report based on discussion of the work? | | Yes |  |
|  |  | |  |  |
| 11. | Are References or Bibliography given at the end of the Report? | | Yes |  |
|  | Have the References been cited properly inside the text of the Report? | | Yes |  |
|  | Are all the references cited in the body of the report | | Yes |  |
|  |  | |  |  |
| 12. | Is the report format and content according to the guidelines? The report | | Yes |  |
|  | should not be a mere printout of a Power Point Presentation, or a user | |  |  |
|  | manual. Source code of software need not be included in the report. | |  |  |
|  |  |  |  |  |

**Page 39 of 40**

