

# CLOUD BASED PLATFORM AS A SERVICES OF PANORAMIC APPROACH OF GRAPHICAL BOUNDARY VALUE DIFFERENTIAL EQUATION SOLUTIONS

S.Satyanarayana<sup>1</sup>, Ch.V.Pani Krishna<sup>2</sup>

<sup>1,2</sup> Associate professor, Department of CSE,  
E-Mail:s.satyans1@kluniversity.in, phanik16@kluniversity.in  
Koneru Lakshmaiah Education Foundation(KLUniversity), Vaddeswaram, India.

## Abstract

*In this paper to derive a graphical boundary value solutions of differential equations using Cloud based platform as a services of R based predictive Analytics tool designed for solving ODEs resulting from 1-D, 2-D and 3-D partial differential equations (PDE) that have been converted to ODEs by numerical differencing this tool Functions that solve initial value problems of a system of first-order ordinary differential equations (ODE), of partial differential equations (PDE), of differential algebraic equations (DAE) and of delay differential equations. In this paper to discuss with Basics of cloud computing and Basics Differential equation and Solving Approach of graphical boundary value solutions of differential equations using cloud based platform as a services of R based predictive Analytics tool*

**Keywords:** cloud computing, PaaS, R programming, predictive Analytics tool, graphical boundary value solutions of differential equations.

## Introduction

### 1. Ordinary differential equations (ODEs)

Differential equations (DEs) occur in many branches of science and technology, and there is a real need to solve them both accurately and efficiently. There are relatively few problems for which an analytic solution can be found, so if we want to solve a large class of problems, then we need to resort to numerical calculations.

#### 1.1 First Order Differential Equations[1]

The general form taken by a first order ordinary differential equation (ODE) is  $y' = f(x, y)$ , where  $f$  is a given function of  $x$  and  $y$  and  $y$  contained in  $R_m$  is a vector. Here  $x$  is called the independent variable and  $y = y(x)$  is the dependent variable.

### 2. Cloud Computing Definition[6]

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics(On-demand self-service, Broad network access., Resource pooling, Rapid elasticity, Measured service), three service models(Software as a Service (SaaS). Platform as a Service (PaaS). Infrastructure as a Service (IaaS)) and four deployment models(Private cloud, Community cloud, Public cloud, Hybrid cloud).

### **3. PaaS [7]**

Platform as a service (PaaS) is a category of cloud computing services that provides a computing platform and a solution stack as a service. Along with software as a service (SaaS) and infrastructure as a service (IaaS), it is a service model of cloud computing. In this model, the consumer creates the software using tools and/or libraries from the provider. The consumer also controls software deployment and configuration settings. The provider provides the networks, servers, storage, and other services that are required to host the consumer's application. PaaS offerings facilitate the deployment of applications without the cost and complexity of buying and managing the underlying hardware and software and provisioning hosting capabilities. There are various types of PaaS vendors; however, all offer application hosting and a deployment environment, along with various integrated services. Services offer varying levels of scalability and maintenance. PaaS offerings may also include facilities for application design, application development, testing, and deployment as well as services such as team collaboration, web service integration, and marshalling, database integration, security, scalability, storage, persistence, state management, application versioning, application instrumentation, and developer community facilitation. Besides the service engineering aspects, PaaS offerings include mechanisms for service management, such as monitoring, workflow management, discovery, reservation, etc..

### **4.Types of PaaS**

#### **4.1 Add-on development facilities**

These facilities allow customization of existing software-as-a-service (SaaS) applications, and in some ways are the equivalent of macro language customization facilities provided with packaged software applications such as Lotus Notes, or Microsoft Word. Often these require PaaS developers and their users to purchase subscriptions to the co-resident SaaS application.

#### **4.2Stand alone development environments**

Stand-alone PaaS environments do not include technical, licensing or financial dependencies on specific SaaS applications or web services, and are intended to provide a generalized development environment.

#### **4.3 Application delivery-only environments**

Delivery-only PaaS offerings do not include development, debugging and test capabilities as part of the service, though they may be supplied offline (via an Eclipse plugin for example). The services provided generally focus on security and on-demand scalability.

#### **4.4 Open platform as a service**

This type of PaaS does not include hosting as such, rather it provides open source software to allow a PaaS provider to run applications. For example, AppScale allows a user to deploy some applications written for Google App Engine to their own servers, providing datastore access from a standard SQL or NoSQL database. Some open platforms let the developer use any programming language, any database, any operating system, any server, etc. to deploy their applications.

#### **4.5 Mobile PaaS (mPaaS)**

The Yankee Group recently identified mobile PaaS (mPaas) as one of its themes for 2014, naming a number of providers including Kinvey, AnyPresence, FeedHenry, FatFractal and Point.io.

## 5. Implementing Initial Value Problems cloud based plat form as a services of R based predictive Analytics tool

**5.1** A simplified form of the syntax for solving ODEs is:

ode(y, times, func, parms, ...)

where times holds the times at which output is wanted, y holds the initial conditions, func is the name of the R function that describes the differential equations, and parms contains the parameter values (or is NULL). Many additional inputs can be provided, e.g. the absolute and relative error tolerances (defaults  $rtol = 1e-6$ ,  $atol = 1e-6$ ), the maximal number of steps (maxsteps), the integration method etc. The default integration method is lsoda.

### 5.2 A Differential Equation Comprising One Variable[2,3,4,5]

Ordinary differential equations are often used in population biology. One of the simplest equations describing population growth is the logistic equation. This equation models the density changes of one species (y) in an environment where competition for available resources reduces population growth at high densities, and eventually leads to negative growth above a specific carrying capacity K. At very low density, the absence of competition allows exponential growth, at a growth rate  $r > 0$ :

$$Y' = rY \left(1 - \frac{Y}{K}\right)$$

$$Y(0) = 12.$$

To implement this IVP in cloud based R we first define the two parameters, r and K, and the initial condition (yini) and assign values to them. The semi-colon “;” separates two statements; the “<-” is the assignment operator. The simple differential equation is implemented in an R function called `derivs` that takes as arguments the current time (t), the value of the dependent variable (y) and a parameter vector (parms), and that returns the derivative, as a list. The parameters r and K, although defined outside of function `derivs`, are also known within the derivative function. We require output at 0.2 daily intervals for 20 days, which we specify in a vector (times); the R function `seq` creates the output time sequence. The model is solved, using R function `ode`.

### 5.3 Cloud based plat form as a services of R Script

solving procedure  $Y' = rY \left(1 - \frac{Y}{K}\right)$

$$Y(0) = 12.$$

```
library(deSolve)
```

```
r <- 1; K <- 10; yini <- 12
```

```
derivs <- function(t, y, parms)
```

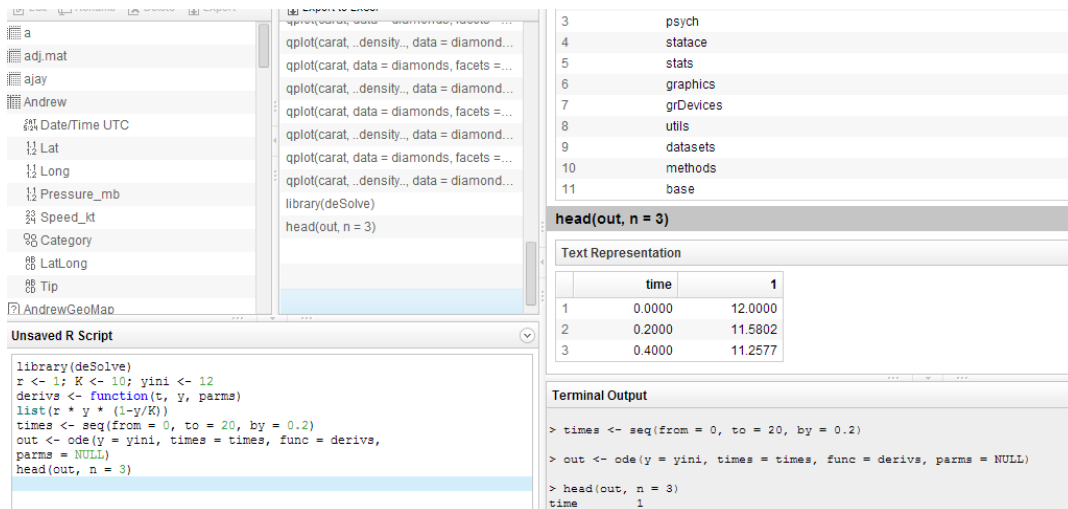
```
list(r * y * (1-y/K))
```

```
times <- seq(from = 0, to = 20, by = 0.2)
```

```
out <- ode(y = yini, times = times, func = derivs,
```

```
parms = NULL)
```

```
head(out, n = 3)
```

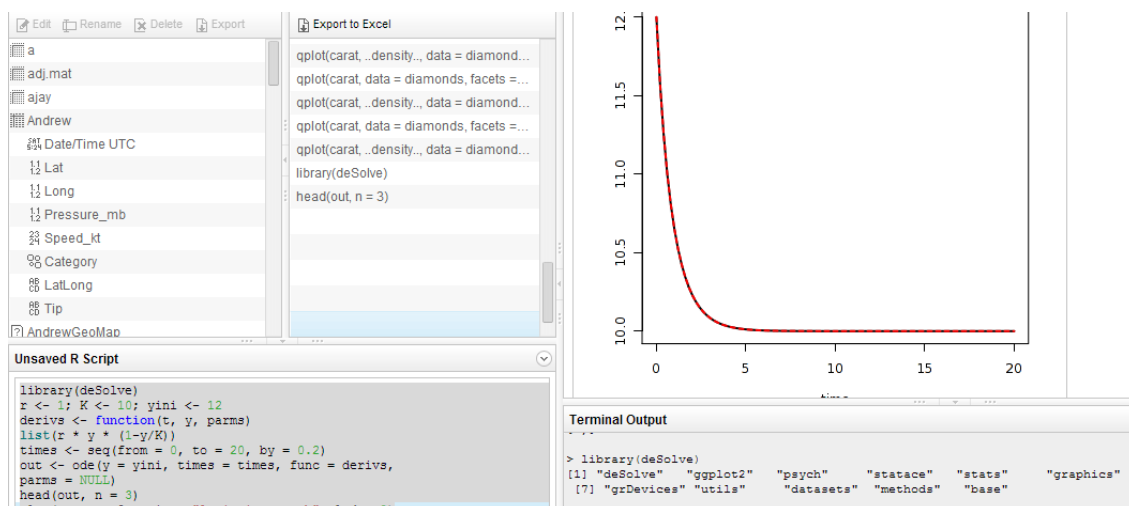


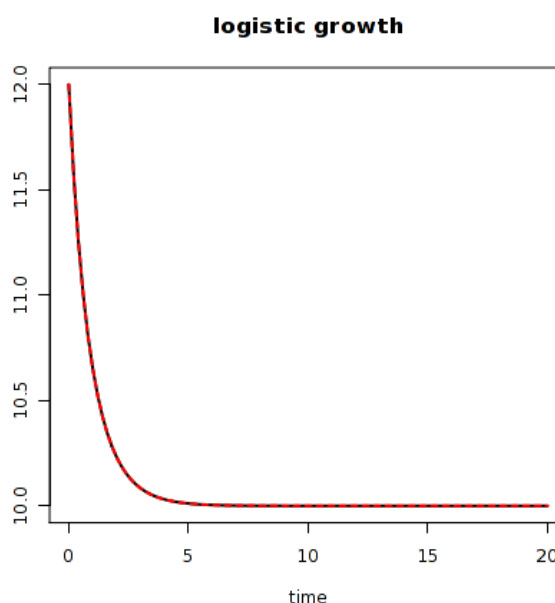
The model output in out is a matrix consisting of two columns, first time, then the state variable value y :

	time	1
1	0.0000	12.0000
2	0.2000	11.5802
3	0.4000	11.2577

#### 5.4 The output of these two solutions is easily plotted, using the deSolve's function plot with the solid lines twice as thick as the default (lwd=2)

```
library(deSolve)
r <- 1; K <- 10; yini <- 12
derivs <- function(t, y, parms)
list(r * y * (1-y/K))
times <- seq(from = 0, to = 20, by = 0.2)
out <- ode(y = yini, times = times, func = derivs,
parms = NULL)
head(out, n = 3)
plot(out, out2, main = "logistic growth", lwd = 2)
```





### Acknowledgements:

We would like to express our thanks to referees for valuable comments that improved the paper. The Author wish to thank Principal & Management, Koneru Lakshmaiah Education Foundation(KLUniversity) Vaddeswaram, India. for their encouragement and cooperation in the preparation of their research paper.

### References

- [1] Brown, P. N., Byrne, G. D., & Hindmarsh, A. C. (1989). VODE, a variable-coefficient ODEsolver. SIAM Journal on Scientific and Statistical Computing, 10, 1038–1051.
- [2] Brugnano, L., Magherini, C., & Mugnai, F. (2006). Blended implicit methods for the numerical solution of DAE problems. Journal of Computational and Applied Mathematics, 189(1–2), 34–50.
- [3] Hairer, E., Norsett, S. P., & Wanner, G. (2009). Solving ordinary differential equations I: Nonstiff problems (2nd rev. ed.). Heidelberg: Springer.
- [4] Soetaert, K., Petzoldt, T., & Setzer, R.W. (2009). R-package deSolve, writing code in compiled languages. Package vignette.
- [5] Soetaert, K., Petzoldt, T., & Setzer, R. W. (2010). Solving differential equations in R: Package deSolve. Journal of Statistical Software, 33(9), 1–25.
- [6] S.Satyanarayana, “CloudComputing:SAAS”, GESJ:Computer Science and Telecommunications, Vol. No. 4(36), pp. 76-79, 2012
- [7] Boniface, M., etc..Platform-as-a-Service Architecture for Real-Time Quality of Service Management in Clouds, Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference ,PP(155 - 160).

Article received: 2014-03-09