www.ijecs.in

International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 7 Issue 6 June 2018, Page No. 23971-23981
Index Copernicus Value (2015): 58.10, 76.25 (2016) DOI: 10.18535/ijecs/v7i6.02

Design and Implementation of Cloud Services by using Python

Priyanka Hariom singh

Abstract:

Cloud Computing DEF: Cloud Computing is a virtualized compute power and storage delivered via platform-agnostic infrastructures of abstracted hardware and software accessed over the Internet. On Client Requirement, IT resources, are generated and willing of efficiently, are dynamically scalable through a verity of programmatic interfaces and are billed variably based on measurable usage. Cloud computing is a general term for anything that involves delivering hosted services over the Internet. These services are broadly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS).

Cloud applications are developed in Platforms as a Service following the PaaS architecture imposed by several providers. We find in the literature some works describing frameworks and architectures for cloud software development by using Python scripting language, but there is a lack of a generic methodology which covers the whole application development lifecycle nowadays Python place Big Roles to maintain Big Data and Best Cloud services to the client.I want to introduce all best methodology to implementing all database, front-end design and maintains and back-end services of PaaS model by using on language i.e. PYTHON.

Keywords: Cloud Computing, Cloud Models, Cloud Providers, DB-Python, Design tools, Python, Analytics, Data Science, Security , PyCharm , RStudio, Azure Container Instances and Azure Logic Apps .

Introduction:

Before Introduction of cloud computing , I would like to Explain Why Cloud need .

Let's Take an example ,you want to host website ,these are the following things that you would need to do.

1. Buy a stack of servers.

2. keeping the peak traffic in mind ,buy more servers.

Disadvantages:

- This setup is expensive.
- Troubleshooting toting problems can be tedious and may conflict with your business goals.
- Since the traffic is varying ,your servers will be idle most of the time.

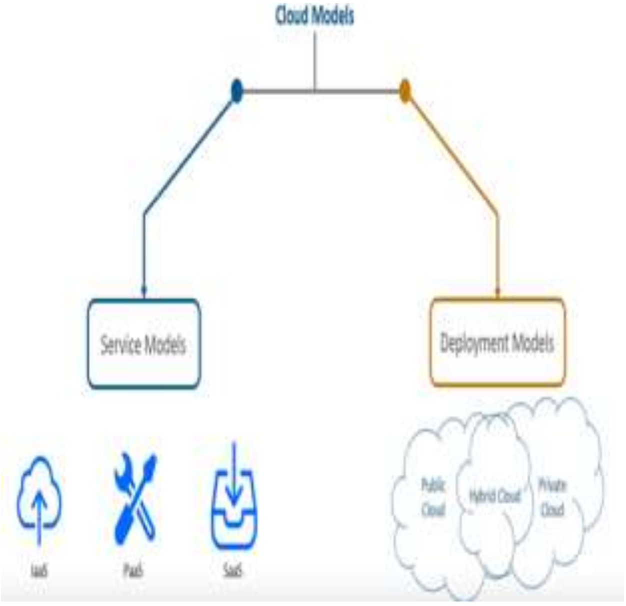
How It Happens Now ?

- Put your data on Cloud Servers and voila!No more buying expensive servers!
- Scalability!Your server capacity will vary according to traffic,how cool is that!
- Cloud providers will manage your all servers,hence don't worry about the underlying infrastucture like Amozon,Azure etc.

What Is Cloud Computing?

It is use of remote servers on the internet to store , manage and process data rather than a local server or your personal computer .Cloud Computing is a virtualized compute power and storage delivered via platform-agnostic infrastructures of abstracted hardware and software accessed over the Internet.

Cloud Models



Cloud computing comes in two categories such as

1. Service Models
2. Deployment Models

1.1Software as a Service (SaaS),

1.2 Infrastructure as a service (IaaS),

1.3 Platform as a Service (PaaS).

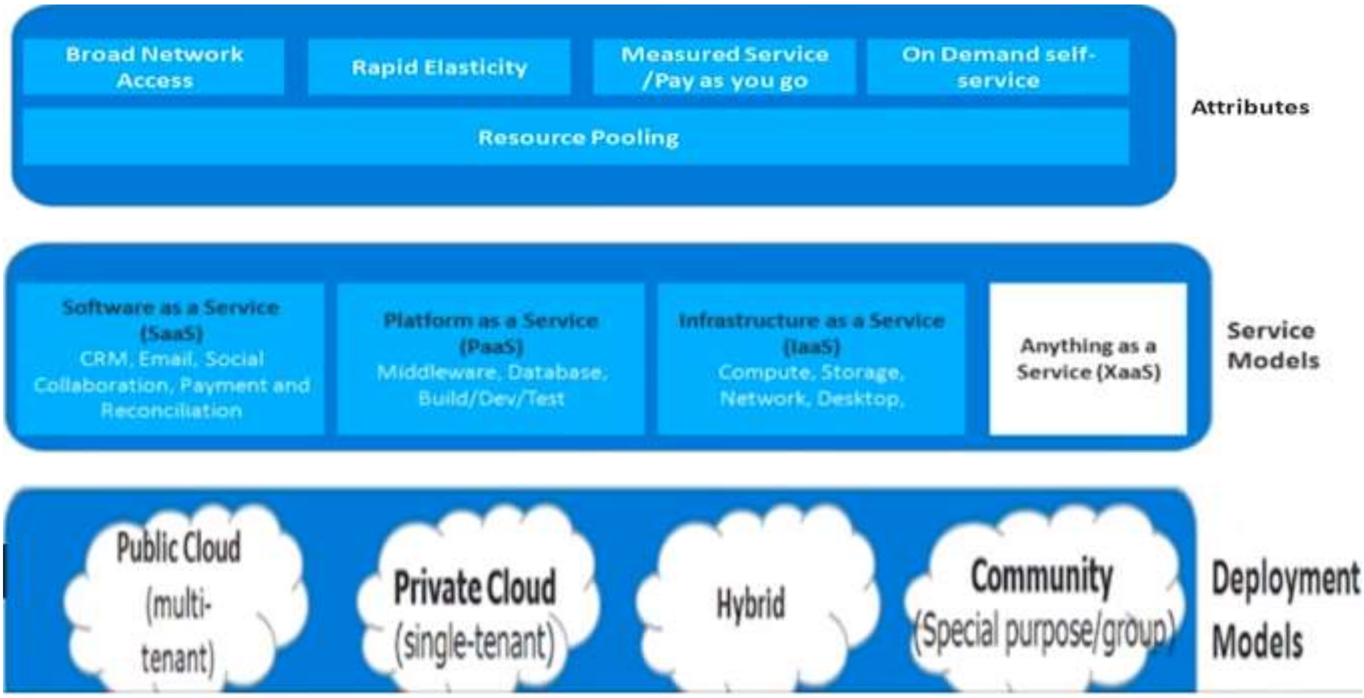
The cloud is available in Three-deployment model namely.

2.1. Public Cloud

2.2. Private Cloud

2.3. Hybrid Cloud

How the Cloud Models Working with Attributes ,Service Models, Deployment Models ?


Broad Network Access
Rapid Elasticity
Measured Service /Pay as you go
On Demand self-service
Resource Pooling
Attributes
Software as a Service (SaaS)
CRM, Email, Social Collaboration, Payment and Reconciliation
Platform as a Service (PaaS)
Middleware, Database, Build/Dev/Test
Infrastructure as a Service (IaaS)
Compute, Storage, Network, Desktop,
Anything as a Service (XaaS)
Service Models
Public Cloud (multi-tenant)
Private Cloud (single-tenant)
Hybrid
Community (Special purpose/group)
Deployment Models

1. Service Models

1.1SaaS(Software as a Service) :

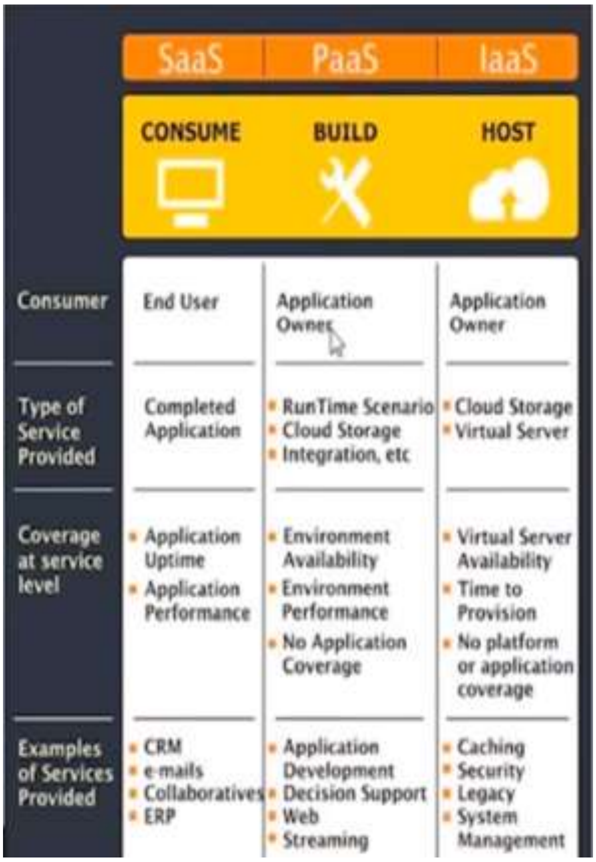
- ✓ The application is hosted centrally.
- ✓ Software testing takes place at a faster rate.
- ✓ Reduction in IT operational costs.
- ✓ No need to install new softwear to release updayes.

1.1 PaaS(Platform as a Servies):

- ✓ Facilitation of hosting capabilities.
- ✓ Designing and developing the application.
- ✓ Integrating web services and databases.
- ✓ Providing Security , Scalability and staorage.

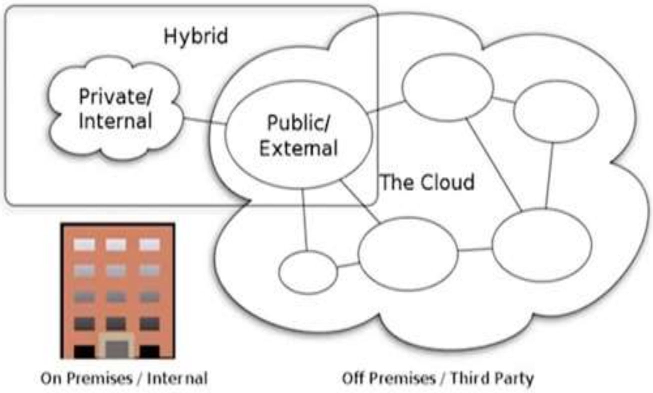
1.2 IaaS(Infrastructure as a Service):

- ✓ Virtualization of Desktop.
- ✓ Internet availability.
- ✓ Use of billing model.
- ✓ Computerized administrative tasks.



	SaaS	PaaS	IaaS
	CONSUME	BUILD	HOST
Consumer	End User	Application Owner	Application Owner
Type of Service Provided	Completed Application	RunTime Scenario Cloud Storage Integration, etc	Cloud Storage Virtual Server
Coverage at service level	Application Uptime Application Performance	Environment Availability Environment Performance No Application Coverage	Virtual Server Availability Time to Provision No platform or application coverage
Examples of Services Provided	CRM e-mails Collaboratives ERP	Application Development Decision Support Web Streaming	Caching Security Legacy System Management

2. Deployment Models



2.1 Public Cloud

- ✓ Mega-Scale Infrastructure Services
- ✓ Sold publically
- ✓ Pay-Par-Use
- ✓ Multitenant applications and services
- ✓ Access virtually unlimited resources

2.2 Private Cloud

- ✓ Finite Infrastructure and Services.
- ✓ Enterprise owned or leased.
- ✓ Charge-back to LOBs or Users.
- ✓ Cloud Computing model in a company’s own datacentre or providers.
- ✓ Access limited resources that must be managed by LOBs or IT.

2.3 Hybrid Cloud

- ✓ Combination of Public and Private Clouds.
- ✓ Mixed usage of both public and private clouds.
- ✓ Enables application components to be spread between multiple public and/or private cloud.

Background

Fig. shows the different- different cloud service models and how they operate to security and user over resources. Logically, as you move up throu layers from IaaS to SaaS, there are fewer security r the first-party (user) and third-party (providers) but at the cost of less control by the use single layer serves a not the same purpose to serv users who are just consistent internet users, as developers. IaaS , PaaS , SaaS Security Control Fig. service models related to security and user control. platform services, also known as Platform as a (PaaS), provide a computing platform or solution s which software can be developed for later deployme cloud. However, there are a lots of security cha because the First party (user)of the cloud have to re third-party companies to offered confidentiality, i and availability.

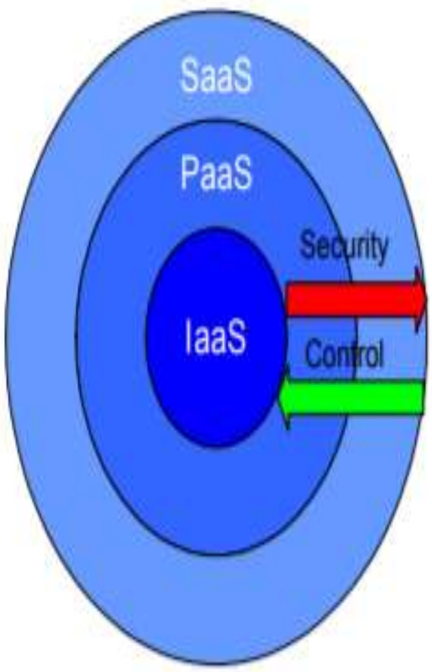


Figure 1. Cloud service models related to security and user control

What is Python?

Python is a high-level, interpreted, interactive and Object-Oriented Scripting language. It is a highly readable language. For example, Python is rummage-sale by more companies (such as Google, Yahoo!, Microsoft, CERN, NASA) and is pragmatic for web development, scientific computing, embedded applications, artificial intelligence, software development, and information security.

Main Features of Python :

- Use of indentation whitespace to indicate blocks
- Object orient paradigm
- Dynamic typing
- Interpreted runtime
- Garbage collected memory management
- a large standard library
- a large repository of third-party libraries

Why cloud platforms should invest in the promise of Python

- Python has emerged as the language developers want to use more than any other for building data-intensive projects.
- Python now powers some of the most complex applications on the cloud Implementation a cloud services by using Python on a PaaS cloud .
- Python having all main features of POP, OOP, java and other languages ,you can say it is more features in one package .

There are many advantages of deploying a python service in a cloud.

According to the 2017 Stack Overflow Annual Developer Survey, Python has emerged as the language developers want to use more than any other this year for building data-intensive projects, which can range from automating robots to fuelling internet of things (IoT) networks with sensor intelligence. Once an obscure scripting language, Python now powers some of the most complex applications on the cloud .Python is on the rise in large part because it allows developers to quickly analyse and organize data, making it especially effective for streaming analytics apps built on the cloud .Streaming analytics, or event stream processing, has become pivotal to the growth of IoT and the corresponding increase of information being collected on the cloud from sensors across industries.

It's the most effective way to keep track of events ranging from equipment failure to financial transactions, as well as to compile and analyse real-time data on the cloud or at the edge of the network.

This compatibility has stimulated a greater number of streaming analytics cloud services to offer the growing Python community the tools needed to compile data and build in their preferred language. In turn, this has accelerated how quickly developers can harness real-time data for more intelligent solutions across manufacturing, finance, operations and other industries . As the Python ecosystem continues to grow, below are some of the top reasons why cloud platforms should invest in Python development, and how cloud infrastructure and analytics services can better power innovations built with Python.

The speedy rise of Python between both inventors and data researchers

In the data science field, there are two main parties—developers and data scientists—that are increasingly converging. It can be seen in the code they use. Most are using either R or Python, coupled with open projects like Spark (for big data) and Tensor Flow (for machine learning), for data processing and analysis. In fast-changing cloud environments, Python has seen increased interest from data scientists for its ease of use and convenience, and for its ability to effectively wrangle data sets, train machine learning models, visualize analytics, and more. For developers, Python is a relatively easy scripting language to grasp—many developers view it as a language with clean syntax and an expansive ecosystem of libraries and tools. In addition to functional programming, Python is also a language that supports object-oriented programming, which provides a quick and consistent method for structuring code. The language's longevity means there is a trove of documentation, which helps questions from developers as they build with it to be answered quickly.

We also can't forget the positive feedback loop created by Python's large and increasingly growing user base. Such a massive user base means those just wading into the world of Python have plenty of resources, including tutorials, code snippets, and its line-up of libraries for machine learning and data analytics. python user base is only going to continue its growth, so it's important for cloud platforms to start investing in building out their compatibility with the language and offering more capabilities and integrations with Python. Both originators and data experts need to work with aggregated data across the cloud in order to fuel streaming analytics and IoT apps, so they're going to need a language with the accessibility and utility that Python offers. Python's expanding potential with cloud services

- With the language growing in popularity, Python has been able to play in new fields within cloud development, such as in many data and IoT services fuelled by cloud infrastructure.For example, Python could be used to build an app on the cloud that can predict the likelihood that a vehicle engine will fail based on its temperature.

- After creating a Python model, which is able to easily be trained on historical failure data, a developer could connect the model into a streaming analytics cloud service to run real-time readings on incoming IoT sensor data, and then quickly configure it to alert the user if engine failure is imminent due to overheating.

- Consider a healthcare team looking to use the cloud to allow doctors to better understand patterns in patients' vital signs.

- Using a combination of Python-based analytics, cloud data services and visualization tools, the development team could easily create an app which would allow the analysis of a patient's data to be instantly visualized—in turn helping doctors to notice key health trends and spikes in a patient more quickly, without having to wait for code to be ingested and translated.

- The future holds only more advancements in how the popular Python language is applied, as we all work together to make it even more accessible and applicable in the cloud.

- If cloud platforms prioritize Python, they can give data scientists and developers access to the new, data-intensive work processes today. Though often perceived as difficult, data science is no longer out of anyone's reach with Python in combination with cloud.

Importantly

- Cloud Computing is enabling more innovation to occur at the developer level than ever before – speed, agility and economics are combining to mean that building a concept product and getting it to market is a relatively painless task Given this fact, we predict a burgeoning number of developers over the years ahead.

- An increasing focus has been placed recently on the data science process; a methodology to govern the enterprise-scale effort that goes into the development, deployment and maintenance of data analytics.

- Data scientists have not been lacking in terms of tools for developing their algorithms but when it comes to deploying their solutions, especially in a hybrid environment, the available tools have not been flexible enough.

- Data researchers affection to write their Python and R scripts exhausting open source tools, such as PyCharm and RStudio. Such tools allow them to work interactively with samples of the data and build the analytics algorithm gradually.

- Maximum of the industrialized data analytics scripts are prerequisite to be deployed in a difficult environment that could span a hybrid cloud/on premises resources.
- Taking the accurate tools to permit the governance and organization of these scripts within a workflow that lights business requirements is a major part of building a success out of the data science effort.
- Microsoft Azure suggestions several services that permit springiness of deployment and combination of data analytics, and allow the algorithms developed by data inventor to be made accessible in a number of ways based on the specific usage situation.
- how to congregation a Python script in Azure Bottle Instances and how to then incorporate the ampoule in a workflow by Azure Logic Apps.
- Using Logic Apps makes it easy to ensure the container is only created on demand and then turned off so that the cost is only incurred when necessary.

Architecture Python script to run:

The Python script to run on demand based on a different-different trigger event (e.g. when new data becomes available).

1. The Python Script retrieves data from an Azure SQL database, operates on the data and then writes the results back to the database as shown in the diagram below.

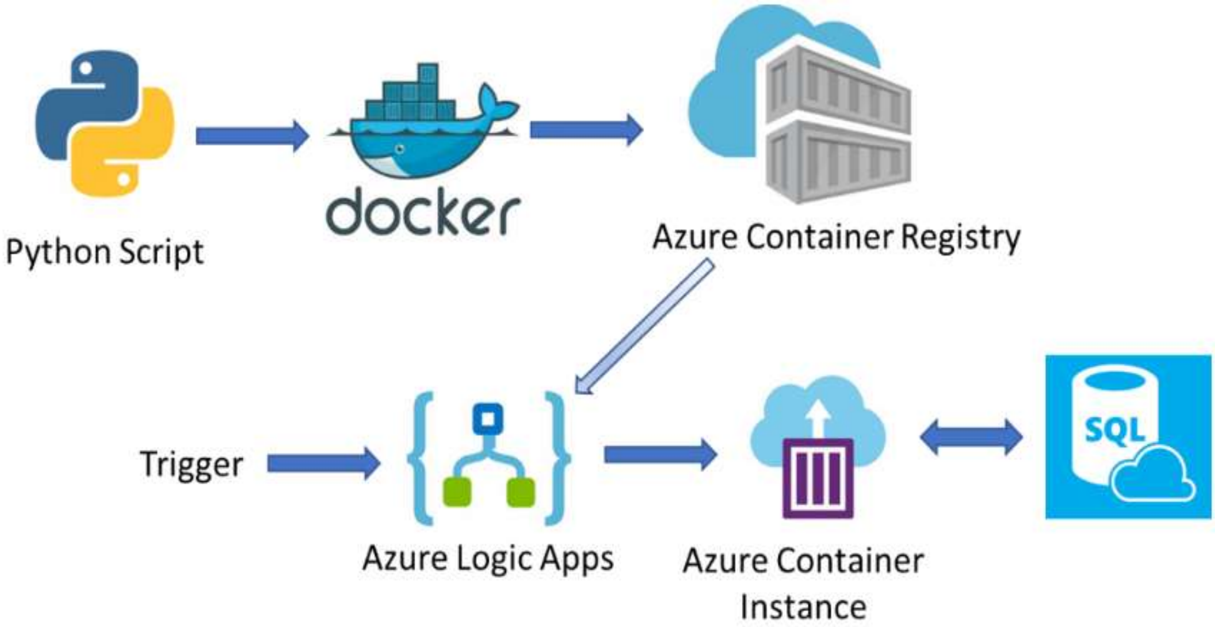


Fig: Architecture Python script to run

2. A Docker container pictorial information (image) hosts the Python script and is registered with the Azure Container Registry.
3. The Logic Apps instance controls the workflow and is instantiated by the trigger signal, creating a container group with a single container based on the image stored in the registry.
4. The container runs the Python script and on completion it is destroyed by the Logic App.

An algorithm of Implementation Of Python Script on Azure Cloud

Step 1:

Use Docker to build a container image that can run the Python script. Since it is necessary for the script to interact with the SQL database, we need to make sure that the Docker file hand-me-down to build the container image holds the essential orientation to the pyodbc library. A whole Dockerfile can be originated here, though you will essential to enhance the compulsory command to include your Python script as the portion of the Dockerfile. For example, to include the Python script "my_script.py" you will need to add the following (note: modify the following two lines based on where you want to place the scripts in the container image):

```
ADD my_script.py /

CMD [ "python", "./my_script.py" ]
```

After you have created your Docker file you can use the Docker Command Line Interface (CLI) "build" command to build your container image:

```
docker build -t <dockerfilename> .
```

Step 2:

Implicates process and uploading the ampule image through the Azure Container Registry, construction definite to tag the image through evidence such as the image version.

You can custom the Azure CLI to accomplish this as described here.

Step 3:

Constructing the workflow exhausting Azure Logic Apps. With the fresh tallying of Container Instance Cluster connectors, Logic Apps can governor the design of a Container Instances confidential container groups, monitor the container state to detect success of accomplishment and then erase the container and the related container group.

Step 4:

By guaranteeing that the container is only energetic for the amount of time essential to wide-ranging the task, charges are minimized.

Step 5:

There are numerous cause types that can be rummage-sale to start the Logic App including web hooks, HTTP announcements, and timed events, permitting the workflow to integrate the Python script execution with external events.

Step 6:

In the Logic App instance exposed in the diagram underneath the produce is set as a timed event.

Step 7:

Once the Logic App obtains the timer event it produces a Container Group and a Container exclusive the group based on the image retrieved from the registry.

Step 8:

A loop is then started that Controls the state of the Container Group till it has prospered (representing that the Python script has done).

Step 9:

The preceding step is to obliterate the Container Group.

Conclusion:

1. Python is billed by the Python Software Foundation as being easy to learn and running everywhere. It's useful for a range of application types, including Web development, scientific computing, and education.
2. Google and Instagram must be between the numerous operators of Python, and the dialectal notches glowing in reputation catalogs. But Python has taken its problems, with queries elevated roughly its performance and enterprise quirks.
3. Most of the established data analytics scripts are mandatory to be deployed in a complex environment that could duration a hybrid cloud /on-premises resources.
4. Having the right tools to allow the governance and deployment of these scripts within a workflow that meets business needs is a major part of making a success out of the data science effort.
5. The growing application has been positioned newly on the data science process; a methodology to govern the enterprise-scale effort that goes into the development, deployment, and maintenance of data analytics. PyCharm, py- Studio tools allow them to work interactively with samples of the data and build the analytics algorithm gradually.

Why python is nowdays best programming language for cloud services implementation and deign.

The Python having lots of resion to choosing as cloud’s programming language

➔**Read it, use it with ease.** Pierre Carbonnelle says,” a Python programmer and blogger who runs the PyPL language index. "This has benefits to you and to others.

➔It helps you think more clearly when writing programs, and it helps the others who will maintain or enhance your program. In both cases, it requires less effort to write a Python program than to write one in another language like C++ or Java." Readability of Python facilitates open source development, Carbonnelle added.

➔ **Python is interpreted language and user friendly** : “Python is easy to use and extremely popular in academia, creating a large talent pool, says Sumit Chachra, CTO at Tivix, a software consulting firm that specializes in Python/Django development”.

➔ **Stephen Deibel says of Wingware**, “Python is a very productive way to write code”, which makes the Wing Python IDE.

➔**Well-designed,Open Soures and Readability:**"Some of this comes from the simple syntax and readability -- there is virtually no 'boilerplate' at all. Some of it comes from the rich, well-designed built-ins and standard library and the availability of many third-party open source libraries and modules”.

➔ Chachra says, Python," By being easy to understand, the code is easier to maintain, is dynamically typed and flexible, with code that is less verbose”.

➔Python, saysHowever, he cites dynamic typing as a potential negative (see below).

1. **Internet of things opportunities.** Python may become popular for the Internet of things, as new platforms such as Raspberry Pi are based on it, Carbonnelle says.
2. Raspberry Pi's documentation cites the language as "a wonderful and powerful programming Raspberry Pi lets you connect your project to the real world."
3. **Asynchronous coding benefits.** Python, Deibel says, "is great for writing asynchronous code, which rather than threading uses a single event loop to do work in small units." This code, he says, is often easier to write and maintain without confusing resource contention, deadlocks, etc.

4. "Python's generators are a great way to interleave running many processing loops in this approach."
5. **Multiparadigm approach bests Java.** Python's programming approach is not as limited as Java's, Carbonnelle says. "For example, you don't need to create an OO class to print 'Hello world' in Python -- you have to in Java." Unlike Java, Python is multiparadigm and supports OO, procedural, and functional programming styles, he says.
6. **Python follow both paradigms oop[Object-Oriented programming] as well as pop[Procedure-oriented programming] language:"** "It's possible to write applications in Python using several programming paradigms, but it does make for writing very clear and understandable object-oriented code."
7. **CPython:** In Python, everything is an object," says Brian Curtin, a member of the Python Software Foundation board of directors and a core contributor to CPython.

Due to lots of features and advantages of python is best for implementation and design cloud services at PaaS model .

End user:

https://cloudmesh.github.io/introduction_to_cloud_computing/class/lesson/linux/python.html

https://www.infoworld.com/article/3233140/python/why-cloud-platforms-should-invest-in-the-promise-of-python.html

file:///D:/cloud%20publication/paas%20on%20cloud.pdf

https://cloudmesh.github.io/introduction_to_cloud_computing/class/lesson/linux/python.html

https://blogs.technet.microsoft.com/uktechnet/2018/04/04/run-your-python-script-on-demand-with-azure-container-instances-and-azure-logic-apps/

https://www.scribd.com/presentation/201379177/Lecture-1-Cloud-Computing-in-Finance

https://www.ibm.com/cloud/info/softlayer-is-now-ibm-cloud?S_TACT=000025WK&S_OFF_CD=10006371&cm_mmc=Search_Google-_-+clouds++server_Broad_&cm_mmca1=&cm_mmca2=&S_PKG=&cm_mmca7=9062279&cm_mmca8=kwd-307859825751&cm_mmca9=738af8c3-5fb6-47b3-91bb-7376494cd3ce&cm_mmca10=269027403771&cm_mmca11=b&mkwid=738af8c3-5fb6-47b3-91bb-7376494cd3ce_1247_60341

https://slideheaven.com/implications-of-security-mechanisms-and-service-level-agreements-slas-of-platfor.html

References

[1] Godin, S., 2008. Tribes: We Need You to Lead Us, 1st ed., Penguin Books.

[2] Brandic I., Broberg J., Buyya R., Shin Y. C. & Venugopal S., 2008. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Available at: Science Direct.

[3] Schwenke, F. & Weideman, M., 2006. The influence that JavaScript™ has on the visibility of a Website to search engines - a pilot study, Information Research Journal, Vol. 11 No. 4.

[4] Oeschger, I. & Turner, D., 2003. Creating XPCOM Components [Online], Available at: http://www.csie.ntu.edu.tw.

[5] Closs L., 2008. XOCOM, Available at: http://www.socialtext.net/.

[6] Preeti, K.S., 2008. Communication between JavaScript and Python using XOCOM (Updated: 22 January 2010) Available at: http://seeta.in/wiki.

[7] Goth, G., 2008. Ultra Low-Cost PCs Redraw the OS Wars, Available at: csdl.computer.org.

[8] Volder, K.D., 2005. JQuery: A Generic Code Browser with a Declarative Configuration Language, Lecture Notes in Computer Science by Springer, Volume 3819/2006, 88-102.

[9] Jonathan, C. & Swedberg, K., 2007. Learning jquery: better interaction design and web development with simple JavaScript techniques, 1st ed., Packt Publishing.

[10] Hulahop Documentation (Retrieved 22 January, 2010), Available at www.packages.debian.org.

[11] Buyya R., Venugopal S. and Yeo C.S., 2008. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities, In: 10th IEEE International Conference on High Performance Computing and Communications (HPCC-08), Sept. 25-27, 2008, Dalian, China.

[12] Bricklin, D., The co-creator of first spreadsheet, VisiCalc [Online]. Available at: http://www.bricklin.com/.

[13] Bletsas, M., Carrano R.C., and Magalhães, L.C.S., 2007. Mesh Networks for Digital Inclusion - Testing OLPC's XO Mesh Implementation. In: 8o Forum Internacional de Software Livre, Porto Alegre.

[14] Google App Engine Framework Documentation (Retrieved 22 January, 2010), Available at http://code.google.com/appengine/docs/Python/overview.html.

[15] Sugar Development Environment Documentation (Retrieved 22 January, 2010), Available at http://wiki.sugarlabs.org/go/Getting_Started.

[16] Brauner J, Foerster T, Schaeffer B and Baranski B, 2009,Towards a Research Agenda for Geoprocessing Services, 12th

[17] Cloud Security Alliance, 2009, Security Guidance for Critical Areas of Focus in Cloud Computing,http://www.cloudsecurityalliance.org (accessed 9 June 2010)

[18] Everett C, 2009, Cloud computing - A question of trust,Computer Fraud & Security, 2009(6), pp. 5-7Google Apps, 2007,Comprehensive review of security and vulnerability protections for Google Apps,www.google.com/a/help/intl/en/admins/pdf/ds_gsa_apps_white aper_0207.pdf (accessed 9 June 2010)

[19] Kaufman LM, 2009, Data Security in the World of CloudComputing, IEEE Security and Privacy, 7(4), pp. 61-64 Knights M, 2009, Can security kill cloud computing?,http://www.itpro.co.uk/610299/can-security-concerns-killcloud-

[20] computing (accessed 9 June 2010)

[21] Ludwig B, 2009, A comparison of PaaS clouds with a detailed reference to security, BSc Hons research eport, University of Pretoria, South Africa, November 2009.

[22] Microsoft Windows Azure, 2009, Azure Services Platform,http://www.microsoft.com/azure/data.mspx (accessed 9 June2010)

[1] Open Cloud Manifesto, 2009, Open Cloud Manifesto,http://www.opencloudmanifesto.org (accessed 9 June 2010)