Jelastic

AJIT BALAGA, S17-IR-2004[1]

[1] *School of Informatics and Computing, Bloomington, IN 47408, U.S.A.*
[*] *Corresponding authors: abalaga@iu.edu, ajit.balaga@gmail.com*

project-000, April 30, 2017

Jelastic (acronym for Java Elastic) is an unlimited PaaS and Container based IaaS within a single platform that provides high availability of applications, automatic vertical and horizontal scaling via containerization to software development clients, enterprise businesses, DevOps, System Admins, Developers, OEMs and web hosting providers.[1]

Keywords: Cloud, I524

https://github.com/argetlam115/sp17-i524/blob/master/paper2/S17-IR-2004/report.pdf

1. INTRODUCTION

Enterprises are moving away from traditional virtualization solutions and transitioning into the cloud. As software development in the enterprise becomes more agile, there is an equivalent demand on IT to provide an infrastructure that is responsive, scalable, highly available and secure. Enterprise IT departments are responding with private cloud and hybrid cloud solutions that provide IT-as-a-Service, a utility approach that delivers an agile infrastructure to the user community, with ultimate control for administrators but self-management for developers and users where appropriate. The promise of the cloud was: simplicity, scalability, availability and reduced operating cost. However, enterprises are quickly finding that current large-scale cloud implementations are often complicated and expensive, often requiring the help of third party integrators. Jelastic is a cloud service that solves the above problems the above promises and allows enterprises to speed up the development process.This paper introduces the architecture behind Jelastic.[2]

2. JELASTIC

Jelastic is a cloud platform solution that combines benefits of both Platform as a Service (PaaS) and Container as a Service cloud models, using an approach unleashes the full potential of a cloud for enterprises, ISVs, hosting service providers and developers. Jelastic software encompasses PaaS functionality, a complete infrastructure, smart orchestration, and containers support - all together.[3]

Jelastic solutions benefits for all kinds of clients:

- enterprises
- hosting providers
- developers

[3]

Some of the Jelastic key benefits:

- a turnkey Platform for Public, Private, Hybrid and Multi-Cloud deployments with automated continuous integration, delivery and upgrade processes
- support of numerous software stacks, extended with cartridges packaging model and custom Docker containers[1]
- automated replication and true automated scaling, both vertical and horizontal - all applications scale up and down on demand
- various development environments for the most comfortable work experience - intuitive UI, open API and SSH access to containers
- intelligent workloads distribution with multi-cloud and multi-region management[4]
- smart pricing integration - alongside multiple billing systems support, Jelastic provides comprehensive billing engine, quotas and access control policies
- embedded troubleshooting tools for metering, monitoring, logging, etc.

[3]

3. ARCHITECTURE

A consistent outline of the underlying Jelastic components with pointers to the corresponding documentation, namely:[5]

3.1. Cloudlet

Cloudlet is a special infrastructure component that equals to 128 MiB of RAM and 400 MHz of CPU power simultaneously. Such high granularity of resources allows the system to allocate the exactly required capacity for each instance in the environment. There are two types of cloudlets:

- Reserved Cloudlets are fixed amount of resources reserved in advance. Reserved cloudlets are used when the application load is permanent.
- Dynamic Cloudlets are added and removed automatically according to the amount of resources required. Dynamic cloudlets are used for applications with variable load or when it cannot be predicted in advance.

[3]

3.2. Container

Container (node) is an isolated virtualized instance, provisioned for software stack handling and placed on a particular hardware node. Each container can be automatically scaled, both vertically and horizontally, making hosting of applications truly flexible. The platform provides certified containers for a lot of commonly used languages and the ability to deploy custom Docker containers. Each container has its own private IP and unique DNS record.[6]

3.3. Layer

Layer (node group) is a set of similar containers in a single environment. There is a set of predefined layers within Jelastic topology wizard for certified containers, such as:[7]

- load balancer (LB)
- compute (CP)
- database (DB)
- data storage (DS)
- cacheVPS
- build node
- extra (custom layer)

[3]

The layers are designed to perform different actions with the same type of containers at once. The nodes can be simultaneously restarted or redeployed, as well as horizontally scaled manually or automatically based on the load triggers, checked for errors in the common logs and stats and make the required configurations via file manager for all containers in a layer. The containers of one layer are distributed across different hardware servers.[8]

3.4. Environment

Environment is a collection of isolated containers for running particular application services. Jelastic provides built-in tools for convenient environment management. There is a number of actions that can be performed for the whole environment, such as stop, start, clone, migrate to another region, share with team members for collaborative work, track resource consumption, etc. Each environment has its own internal 3rd level domain name by default. A custom external domain can be easily bound or even further swapped with another environment for traffic redirection.[1]

3.5. Application

Application is a combination of environments for running one project. A simple application with one or two stacks can be run inside a single environment. Applications with more complex topology usually require more flexibility during deploy or update processes They may be distributed across different types of servers and several environments, to be maintained independently. Application source code can be deployed from:[4]

- GIT/SVN repository
- local archive
- custom Docker template

[3]

3.6. Hardware Node

Hardware node is a physical server or a big virtual machine that is virtualized via KVM, ESXi, Hyper-V, etc. Hardware nodes are sliced into small isolated containers that are used to build environments. Such partition provides the industry-leading multitenancy, as well as high density and smart resource utilization with the help of containers distribution according to the load across hardware nodes.[5][7]

3.7. Environment Region

Environment region is a set of hardware nodes orchestrated within a single isolated network. Each environment region has its own capacity in a specific data centre, predefined pool of private and public IP addresses and [4] corresponding resource pricing. Moreover, the initially chosen location can be effortlessly changed by migrating the project between available regions.[6]

3.8. Jelastic Platform

Jelastic Platform is a group of environment regions and cluster orchestrator to control and act like a single system. This provides versatile possibilities to develop, deploy, test, run, debug and maintain applications due to the multiple options while selecting hardware - different capacity, pricing, location, etc. The platform provides a multi-data center or even multi-cloud solution for running your applications within a single panel, where each Platform is maintained by a separate hosting service provider with its local support team.[3]

4. CONCLUSION

For enterprises, moving from traditional virtualization to the cloud using PaaS and IaaS can be a daunting proposition. However, the cloud market is growing rapidly and enterprises are recognizing that PaaS allows them to develop and deploy scalable, highly available cloud-based applications in a rapid and agile fashion. Enterprises can capitalize on this new and sticky revenue stream by quickly implementing PaaS and establishing a brand-defining presence in the market. Jelastic provides the only integrated private cloud solution that integrates PaaS/IaaS and is specifically built for enterprises.[2]

REFERENCES

[1] "Jelastic," Web page, Dec. 2016, page Version ID: 754931676. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Jelastic&oldid=754931676

[2] "5 Key Reasons Why Jelastic PaaS and IaaS for Private Cloud is Better," Web page, Jul. 2014. [Online]. Available: http://blog.jelastic.com/2014/07/10/why-you-should-use-jelastic-platform-as-infrastructure-for-private-cloud/

[3] "Jelastic Multi-Cloud PaaS and CaaS for Business," Web page. [Online]. Available: https://jelastic.com/

[4] S. Yangui and S. Tata, "Cloudserv: Paas resources provisioning for service-based applications," in *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*. IEEE, 2013, pp. 522–529. [Online]. Available: http://ieeexplore.ieee.org/abstract/document/6531799/

[5] "Jelastic Cloud Union Catalog: Choose Your Service Provider," Web page. [Online]. Available: https://jelastic.cloud/

[6] "Auto scaling Jelastic PaaS in UK, USA Southwest, and Singapore," Web page. [Online]. Available: https://www.layershift.com/jelastic

[7] J. Chavarriaga, C. A. Noguera, R. Casallas, and V. Jonckers, "Architectural tactics support in cloud computing providers: the jelastic case," in *Proceedings of the 10th international ACM Sigsoft conference on Quality of software architectures*. ACM, 2014, pp. 13–22. [Online]. Available: http://dl.acm.org/citation.cfm?id=2602580

[8] "Jelastic PaaS Cloud Application Hosting," Web page. [Online]. Available: https://www.lunacloud.com/cloud-jelastic