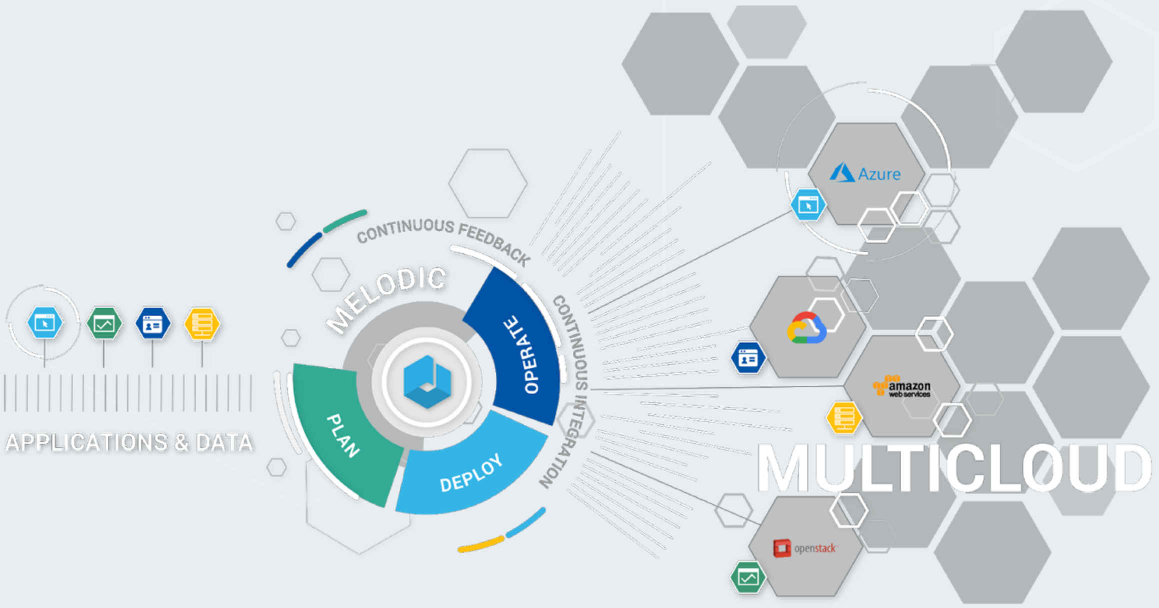
Melodic
optimized multicloud

Multicloud Optimization Platform

STRATEGY AND METHODOLOGY FOR THE MULTI-CLOUD DEPLOYMENTS

KATARZYNA MATERKA, MICHAŁ SEMCZUK



MULTI-CLOUD APPROACH AND CHALLENGES

Cloud Computing[1] is already a basic model of implementing new applications and solutions, at least in the most technologically advanced countries. The overwhelming majority of new applications is designed as native cloud applications, in order to maximize the effects possible thanks to the cloud approach, i.e. the speed of application development and cost savings.

However, with the wider and wider use of cloud solutions, it has become increasingly important to transfer loads between cloud environments of different vendors – Multi-Cloud approach - as well as to create applications in such a manner that they can be automatically deployed in different environments.

This approach brings significant profits to organizations that use cloud solutions. It allows them to choose the best service provider for a given application, and even - virtually on an ongoing basis - optimize costs, increase security and stability of the solution by diversifying contractors.

[1] https://www.techopedia.com/definition/2/cloud-computing

However, on the other hand, cloud solution providers have a very diversified offer, different ways of using it (API[2]) and various, incomparable price lists. In addition, it is in the interest of suppliers to attach the client as much as possible to each other and prevent him from moving to another supplier (vendor lock-in[3]). Therefore, applying the Multi-Cloud approach requires solving the following problems:

- Preparation of application implementation instructions in an irrespective manner of Cloud Provider
- Automatic implementation of transparent applications in various cloud providers
- Selection (optimization) of the best combination of offers from different cloud providers, depending on the application's requirements.

MELODIC – MULTI-CLOUD OPTIMIZATION PLATFORM

The above problems are solved by the MELODIC platform[4]. The platform has been built as a part of the Programme Horizon 2020[5] project with the same name and is being developed as the open source model. The commercial quality has been achieved thanks to the company 7bulls.com, which also offers a set of services related to this platform.

MELODIC allows, unlike solutions supporting only containers such as Kubernetes[6], for implementing applications that support various types of components: virtual machines, containers, big data (Spark[7]) platforms and serverless components. MELODIC is integrated with leading cloud service providers: AWS[8], Azure[9] and Google Cloud Platform[10], as well as with service providers using OpenStack[11]. Integration with local cloud providers is also being carried out. In addition, unlike Kubernetes, which only allows you to optimize resources within an existing cluster, MELODIC allows you to dynamically add cloud resources as the application needs, as well as delete them when they are not needed. As a result, the use of

[2] https://www.techopedia.com/definition/24407/application-programming-interface-api

[3] https://www.techopedia.com/definition/26802/vendor-lock-in

[4] https://www.melodic.cloud/

[5] https://ec.europa.eu/programmes/horizon2020/

[6] https://kubernetes.io/

[7] https://spark.apache.org/

[8] https://aws.amazon.com/

[9] https://azure.microsoft.com

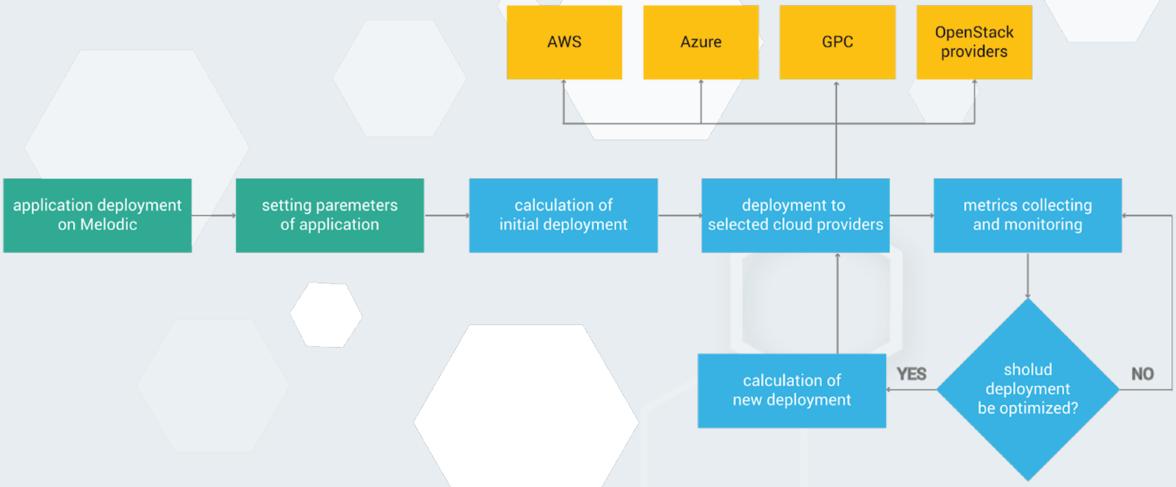
[10] https://cloud.google.com/

[11] https://www.openstack.org/

resources is tailored to the needs of the application and there is no need to maintain an oversized infrastructure.

The key element of the platform is based on the TOSCA standard (Topology and Orchestration Specification for Cloud Applications[12]) is the CAMEL language (Cloud Application Modeling and Execution Language[13]), allowing to describe app requirements and infrastructure independently of a specific supplier, as well as selecting the most optimal implementation model depending on the characteristics of the application. An additional element of the platform is the ability to optimize Big Data solutions and data locality awareness.

Figure 1 Deployment process


AWS
Azure
GPC
OpenStack providers
application deployment on Melodic
setting paremeters of application
calculation of initial deployment
deployment to selected cloud providers
metrics collecting and monitoring
sholud deployment be optimized?
YES
NO
calculation of new deployment

The process of modeling/describing applications in CAMEL language within the Melodic platform first includes defining its components, connections between them, as well as requirements regarding the performance and resources, along with the way of implementing the application. In the next step, the implementation configuration is automatically optimized - the platform "decides" which and where the infrastructure should be used. Initial

[12] https://searchcloudcomputing.techtarget.com/definition/TOSCA-Topology-and-Orchestration-Specification-for-Cloud-Applications

[13] http://camel-dsl.org/

optimization is made on the basis of parameters specified by the user. Optimization is one of the strongest points of the platform. Advanced methods based on Constraint Programming and Reinforcement Learning (Stochastic Learning Automata) are used to solve the optimization problems. MELODIC also includes a unique module for assessing the usability of a business implementation, based on an adaptive usability function.

On the basis of a specific, optimal configuration, a precisely defined infrastructure is automatically created for selected cloud service providers (virtual machines with set parameters) and then the application is implemented along with connection settings between components.

After implementation, the application is monitored – collected are, among others, metrics defining the characteristics of its operation, which at the same time constitute the basis for automatic optimization of application implementation, basing on the current values of the metrics.

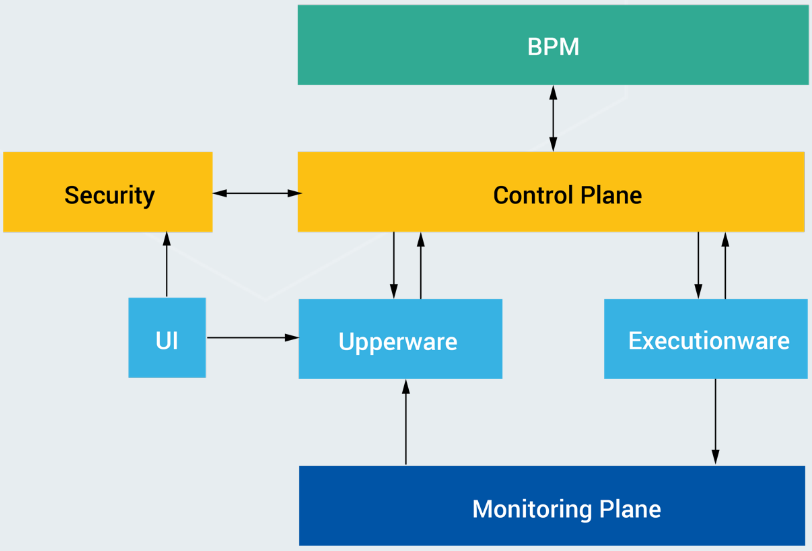
MELODIC ARCHITECTURE

The MELODIC platform consists of the following modules:

- Upperware module: this module is responsible for converting the user's CAMEL model characterizing its application into an implementation plan using application deployment reasoning. He is also responsible for the orchestration of this plan using the Executionware module. The Upperware software is also responsible for detecting the possibilities of user application reconfiguration and their enforcement. This includes both local and global reconfiguration, the former being in the form of performing specific scalability activities in the context of running specific scalability rules.
- Executionware module: supports general resource management and is responsible for performing each step related to the application deployment plan. In the context of implementing the application component, this module is also responsible for managing the life cycle of application components and installing sensors that are useful for monitoring them.
- User Interface module: includes editors and user interfaces that can help him use the most important facilities and services offered by the platform.

- Security module: includes some security services that focus on enforcing controlled access to platform resources as well as proper management of the credentials in the cloud. These services are well integrated with the rest of the platform modules and platforms.

Figure 2 High Level Architecture (HLA) of MELODIC platform



BENEFITS OF USING MELODIC

Below are presented benefits of using MELODIC platform for different types of applications and various domains. Also, both, the cloud native and legacy applications are presented, and benefits are described. The benefits are not only cost savings related, but there are also gains in terms of performance, security and reliability.

Digital Assets Management system – Kuiper DAM

Kuiper DAM[14] is a digital asset management system built using standard master-worker pattern. System is deployed on AWS cloud and using Auto scaling of VMs feature of AWS to implement horizontal scaling. The complete installation of Kuiper DAM contains about 10 virtual machines, type m5.large on AWS, two instances of ElasticSearch cluster and 10 SQS queue. In case of higher workload up to 30 new instances of workers application (each on

[14] https://www.7bulls.com/en/product/Kuiper%20DAM

dedicated VM) is started up. After the increased workload is processed the auto scaling instances are terminated. The payment is only for real time of existence of auto scaled virtual machines.

Using MELODIC we introduce the capabilities of serverless deployment and multi-cloud deployment. The deployment of the core instances (VMs) of all environments of the Kuiper DAM is done using cheaper cloud service provider (Hetzner[15] or Aruba[16]), only SQS, ElasticSearch are used from AWS. The autoscaling feature will be handled by serverless components instead of cloning virtual machines. Thanks to that we have benefits of possibility of unlimited horizontal scaling using serverless components (with payment only for real usage) and much cheaper installation of core components.

So final multi clouds configuration will be looking as follows:

- 10 VMs moved from AWS to Hetzner or Aruba
- 10 SQS queue stayed on AWS
- 2 instances of Elasticsearch stayed on AWS
- Autoscaling using serverless components

CALCULATIONS OF BENEFITS

The VMs on Aruba are 6 times cheaper than AWS ones, so the cost per VM (estimated) is lowered from 150 USD on AWS to 25 USD on Aruba. Thanks to that monthly cost of operating systems (VMs part) could be lowered from 1500 USD to 250 USD. Using serverless components instead of VM could lower cost approximately 500 USD monthly (30 auto scaled machines monthly costs 860 USD comparing to 350 USD cost of similar serverless components). So, the 3 years TCO could be lowered from 85 kUSD (VMs part) to 21.6 kUSD. Savings 63.4 kUSD (for one, medium size system).

The system is deployed on two different cloud providers, so in case of failure of one provider the second one could be used to fully operate systems (needs to have Elastic search also installed on Aruba).

Thanks to usage of MELODIC this optimization could be done automatically, real time, based on actual price estimation for cloud providers. Also using MELODIC orchestration feature the

[15] https://www.hetzner.com/cloud

[16] https://www.arubacloud.com/

usage of many cloud providers adds only minimal overhead to management and deployment tasks for IT operation team. Estimated effort for migration to this model for DAM system including changing to serverless components is up to 50 MD (cost ~ 25 kUSD).

OPTIMIZATION OF CLOUD USAGE FOR BIG COMPANY (BASED ON POLISH TELECOM)

Typical big company is using over 10 000 virtual machines various sizes (except many bare metal servers). The VMs are usually created by hand, at least using some configuration scripts like Ansible[17] or Chef[18] and after installation there are no control on usage. Very often VMs are oversized.

Using MELODIC capabilities of automatic deployment and creation of VMs based on real application requirements it will be possible to optimize such deployments to use cheaper cloud providers and/or optimize configuration of VMs to lower their size.

We estimate that thanks to using MELODIC numbers listed below could be achieved (pessimistic calculation):

- 10% of VM could be replaced by serverless components.
- 10% of resources could be backed-up and removed due to non-usage.
- 10% of resources could be scaled down (to approximately 50% of original size).
- 20% of resources could be migrated to cheaper provider.

CALCULATIONS OF BENEFITS

Assuming that typical VM size is 4 cores and 16 GB RAM with 500 GB storage and based on the numbers listed in previous chapter and assuming that the monthly cost of that VM is 160 USD monthly (half of the price of AWS) benefits of deploying MELODIC for such type of organization will be as follow:

- 90 kUSD monthly due to change cloud provider (20% * 10000 VMS * 45 USD), assuming saving 45 USD monthly on each VM

[17] https://www.ansible.com/

[18] https://www.chef.io/

- 80 kUSD monthly due to limiting size of machines (10% * 10000 VMS * 80 USD)

Total monthly savings will be around 436 kUSD, yearly savings will be around 5 mln USD, 3 years savings will be 15 mln USD.
Estimated cost of implementation of MELODIC for big organization is ca. 1500 MD (300 MD - as is analysis, 600 MD preparation of CAMEL description for already deployed VM's and systems, 180 MD initial deployment, 330 MD testing of deployment) which cost ~ 500 kUSD.

HYBRID USAGE OF CLOUD (BASED ON RETAIL E-COMMERCE CASE)

This scenario described real e-commerce retail architecture and usage, based on one of the polish biggest e-commerce sites.
The company is an owner of big e-commerce portal. Their usage pattern is as follows:

- Q1-Q3 usage is around 30% of maximum, with some peaks around 50% maximum.
- Q4 usage is three times bigger on average (mostly in December) than yearly average.
- The infrastructure of e-commerce solution is build using VM ware solutions and hosted on own infrastructure.
- The number of production VMs (not counting testing and dev env) is around 100 VMs (web servers, appservers, cache and db server)
- 160 kUSD monthly due to removing spare resources (10% * 10000 VMS * 160 USD)
- To handle average traffic there is a need for around 30 VMS.

Due to very different usage in q4 against q1-q3, the infrastructure could be changed to move to cloud computing 70% of virtual machines (70 machines). Thanks to traffic for the most part of the year could be handled using its own resources and only in q4 (or even only in December) additional resources could be used in the cloud. There will be no need to invest in own over-sized hardware platform which is used only in one month of the year.

CALCULATIONS OF BENEFITS

Assuming the monthly cost of one VM machine around 160 USD the yearly benefits of this move could be 123.2 kUSD (70 VMS * 160 USD * 11 months), three years savings 369.6 kUSD. Additional benefit is unlimited and flexible scalability in case of higher than expected workload. More benefits could be achieved using different architecture approach and serverless computing.

Estimated cost of implementation of MELODIC for big organization is ca. 150 MD (30 MD - as is analysis, 70 MD preparation of CAMEL description for already deployed VMs and systems, 10 MD initial deployment, 40 MD testing of deployment) which cost ~ 50 kUSD.

- 106 kUSD monthly due to migrate to serverless components (10% * 10000 VMS * 104 USD) - typical VM costs 160 USD monthly with serverless components costs around 0,3 of the cost of this VM.