# Software as a Service operation model in cloud based ERP systems

**István Orosz, Attila Selmeci**

**Abstract:** Cloud based operation model has become a new standard in ERP system implementation. This has brought a new software abstraction layer over the already existing datacentre model, which covered all the middleware, operating system and database level architecture elements. This new abstraction layer covers the complexity of the middleware and below layers, making it possible for the owners to deal with only implementing the core business logic. There is always a risk that more resources are allocated to the infrastructure side, using this model resources can be reallocated back to the business side. Overall this could lead to longer software lifecycle, because the core logic is separated from the rapidly changing implementation layer. The standard release management, which covered the way from pre alpha state to the gold release, was substituted in this model with a continuous release process. This leads to the possibility of using the latest version of the software always in the Software as a Service operating model. The code reusability (and refactoring) is also modified, having an abstraction layer between the platform independent model and the platform specific model. The transformation layer between these two has to be able to ensure the long term reliability of the software product. Having a proper transformation layer keeps the roles separated, the end user do not have to build up an IT support background, because this layer hides the infrastructure questions. This paper focuses on the change management procedures when change the operation environment from the standard datacentre solution to the cloud based SaaS model.

This paper describes the different directions which can be used for an ERP environment offered by cloud technology, and focuses on the diverse change management procedures on switching the operation environment from the standard on premise datacenter solution to the cloud based SaaS model.

**Keywords:** cloud base operation model, ERP, SaaS, model driven architecture, change management, code reuse

## Introduction

Nowadays the cloud-based solutions are very popular, but not always understood. Many companies would like to use it, but do not know how to start, for what use it in the current solutions. There are several option offered by ERP software vendors and cloud providers to own bigger market share. If we consider the ERP offerings, many services were developed originally for cloud, some others are enabled to be provided on cloud, not only as on premise solutions. There are some efforts to rebuild existing, proven solutions for cloud usage as well.

Standard development and release strategy can be grouped in different categories, like: Diagnostic, Analysis, Design, Development, Deployment and Operation [1]. In the datacentre based operational model, this required the close interaction of the organization, for the whole software lifecycle. When migrating or upgrading software solutions for cloud based operation, these phases will act differently.

The cloud-based operation cannot be separated from the provided services. The two of them act together as a new abstraction layer for software solution, where layers responsible for the implementation are strictly segregated from layers, which are responsible for modeling the core business logic.

The article is structured as follows. The first section gives a definition for the cloud based ERP operation models focusing on Software as a Service and then gives a brief review on the impact of information technology applications. Then the importance of code reusability (and refactoring) is demonstrated through an available tool in Microsoft Dynamics AX environment, for checking the integrity and consistency of code relevant changes and change management. On the other hand the popular SAP solutions are also mentioned by examples and implementation difficulties.

## Service based operation in cloud environments

*The rest of the paper describes the work that has been done. The expected length of the paper (with the references included) is four pages, please fill, but do not exceed this.*

There are more service based operation models in cloud environment, according to what kind of service level is hidden from the business. There are three main operation models [2]:

- **Infrastructure as a Service (IaaS):** provides the lowest set of hosting services. The service level takes care of the virtualization, servers, storage and networking, so almost the infrastructure-like operations.
- **Platform as a Service (PaaS):** one step over the IaaS, this level additionally handles the operating system, middleware interfaces and the runtime software modules.
- **Software as a Service (SaaS):** over the PaaS, the customer has to deal with modellng the business processes implemented by the software solution, and can use the whole underlying infrastructure as a cloud service.

The services can be offered only if there are redundancy, virtualization, management and automation on many levels or layers of infrastructure and applications. From technical perspective (as a hardware oriented view) the management and automation raise the cloud above the known virtualization techniques. From application perspective we have to consider that services can be offered if they exist as separated units and they can be connected directly or in the background. We are also referring to cloud-based solutions in case of application running in a cloud environment. These are not really SaaS examples, but rather application on PaaS or even IaaS bases. The cloud environment what we use can be different:

- **Public clouds:** service providers offer their resources to the public.
- **Private clouds:** designed for exclusive use of a single organization (mainly local).
- **Hybrid clouds:** combining the two previous, part of the infrastructure runs in private, the remaining provides public services.

VPC is another layer over the public clouds, engaging a Virtual Private Network (VPN) protocol, which enables service providers to design and use their own network and security. For more customers it is short transition from datacentre architecture to cloud, while still owning the virtualized network layer [2]. In our examples we show the differences of the solutions offered by larger ERP provides in almost each cases.

## Requirements and possibilities in layers

The layered architecture of the cloud offers more control over the responsibility, whereas every layer owner has to focus on the specific services provided their layer. Resource planning is more flexible, because common pools of computing resources can be dynamically assigned to consumers. This is one of the key properties of cloud computing, allowing resource allocation on the fly. When analysing cloud based service, one of the key point is on demand resource allocation. Cloud operation has to satisfy the service level objectives (SLO), with the possible lowest operational cost. The goal is to optimize the mapping of SLOs (like Quality of Service, QoS) to low level resources like CPU and memory usage. Automatized resource distribution for internet based application were studied in the past[3]. The performance model typically based on: a) number of application instances which are required to handle the requirements which satisfies QoS b) predicting future demand and resource needs c) automated resource allocation based on predictions on the future demand. There are several ways to build up these performance model, like statistical machine learning[4] or deep learning algorithm.

This leads to resource usage based pricing, as charging the customer per use bases. Clouds are based on service oriented operating model, so it natural to provide a consistent service level agreement (SLA). Clouds are also frequently based on geologically different located data centres, which can minimize the risk of service outage. Optimally consolidating server in a data centre often described as a variant of the vector bin-packing problem[5] that is an NP-hard problem of optimization. These kinds of activity must affect performance in any way. Shared resource between servers (network, disk storage, etc.) allocation can lead to traffic jam amongst servers, when VM changes its configuration[6].

## Possible solutions for virtualization and clouds under ERP services

There are many virtualization theorems and implementations available. [7] The Microsoft environment does not really support the application virtualization (like multi node clustering), though technically the operating system is capable to handle the tasks. In this kind of virtualization the services

always hold their resources (mainly IP addresses and names, storage areas, and user environments). This makes the application (or service) capable to be moved from a host to another one without having configuration problems in the consuming services, application. Main disadvantage of this kind of virtualization is that the service should be stopped before move (like in a HA-cluster solution). The other, nowadays used virtualization model is the kernel level sharing, so-called light way virtualization. It has some history as well because earlier the SUN Microsystems (zone) and IBM (AIX wPAR - Workload Partitioning) implemented already this feature, where isolations are on kernel namespace and control group level. The Linux operating systems offer the LXC (Linux Container) as a solution for that. The Docker solution expanses this feature, by managing the OS container capabilities. It can now manage Windows based containers as well, but the base operating system should be Windows certainly.[8] The last, well-known virtualization is the hypervisor technique. The most popular implementations are the VMware and Hyper-V. One of the main differences between the light way virtualization and hypervisor virtualization is the overhead what the hypervisor layer gives to the environment.

ERP systems as on premise solutions can exploit almost each virtualization techniques, which is good for using basic private cloud environments. For SaaS usage these szstems should have been refactored for reusable pieces as services.

## ERP solutions in practice

In our paper we considered two large ERP vendors: Microsoft and SAP. Both vendors have their own on premise solutions, which can run in cloud environment using the IaaS or PaaS types. SAP bought some cloud-based solutions in the last years offering real cloud services (SaaS): Hybris, SuccessFactor or FieldGlass. Around the ERP II. era the Service Oriented Architecture (SOA) was the goal for each vendor or at least to offer a possible enabler for making service orchestration, having a service repository and enabling high level workflows above the services. This was one of the main drivers to the service offering on cloud. The ERP vendors tried to separate and degrade the modules and applications to smaller pieces, which could be simple services. Unfortunately this direction did not bring the expectations. The cloud readiness and the mobility forced the companies to create real service offerings.[9] Microsoft has reconstructed the Dynamics as described later. SAP started this reconstruction, but it is still not fully completed, but some features are available already.

In the practice many companies are still afraid of using cloud technologies, especially public cloud solutions. The thinking, internal functioning is not yet ready for that. Many regulations should also be considered regarding the data sensitivity and location. These companies are thinking about building an own private cloud or using hybrid cloud as a starting point. The public cloud providers have a huge advantage by having the working management and automation toolset for IaaS and PaaS. Some cloud partners offer parallel PaaS and SaaS possibilities by giving add-ons for installing, deploying ERP components on the public cloud. If a company would like to build their own private cloud, it should employ some good experts, or buy consultancy. For long-term investment it can be a good direction.

As the first step the companies try to build a cloud environment (mainly IaaS) for disaster recovery (DR) purposes. They do not consider exactly the costs of moving the production to the DR site located in the cloud and the cost of coming back to the normal on premise functionality. The other main aspect is that it is better to have everything on the same side as using hybrid solution. If more of the applications are on the cloud, the ERP can be implemented there as well.

## Additional aspects

Components based software development processes, based on the OOP methodology, supports code reuse strongly. These technologies, like CORBA, J2EE, COM+, .NET, offers a middleware layer, where reusable components can be built. Change management methodologies are need to be revised as the number of companies going into cloud based service oriented operational models are rapidly growing. The technology changes trigger organizational, business operations and software development side changes also. New predefined SLAs are needed for audit trails and software change management. The security structure is also more complex in an agile cloud environment. Most SaaS cloud implementation are based on existing IaaS cloud implementations, for VM and application hosting.

SaaS ERP services are now available for public and private cloud environments as well.

## References


[1] Richard Murch. *The Software Development Lifecycle - A Complete Guide* Amazon Digital Services LLC, ASIN: B007ZCRP1I

[2] Zhang, Q., Cheng, L. & Boutaba, R. J. *Internet Serv Appl* 1: 7. doi:10.1007/s13174-010-0007-6 (2010)

[3] Zhang Q et al. *A regression-based analytic model for dynamic resource provisioning of multi-tier applications* In: Proc ICAC (2007)

[4] Bodik P et al. *Statistical machine learning makes automatic control practical for Internet datacenters* In: Proc HotCloud (2009)

[5] Chekuri C, Khanna S. *On multi-dimensional packing problems* SIAM J Comput 33(4):837-851 (2004)

[6] Padala P, Hou K-Y et al. *Automated control of multiple virtualized resources* In: Proc of EuroSys (2009)

[7] A. Selmeci, T. Orosz, Gy. Györök. *Innovative ERP virtualization* ISBN 978-1-4799-0303-0 In: IEEE 11th International Symposium ib Intelligent Systems and Informatics (SISY), 2013, Subotica Serbia, pp. 69-75.

[8] MS Contributors. *MS Web Document* https://docs.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/devenv-running-container-development

[9] A. Selmeci, T. Orosz, Gy. Györök. *Teaching ERP User Interfaces: Adequate Sequences of Topics and Technologies* ISBN: 978-1-4673-8739-2 In: IEEE 14th Jubilee International Symposium on Applied Machine Intelligence and Informatics (SAMI), 2016, Herlany Szlovákia, 7 oldal