

# Saving Energy and Resources Through Cloud Computing

22 March 2013 | **Digital Services Innovation Articles, Software Engineering** | [Michael Maurer](#), [Roman Bertolami](#)

**Reading time:** 4 minutes

**Dealing with energy wisely is a cross-cutting challenge worldwide, also for IT, and especially for Cloud Computing. I finished my PhD at TU Vienna last year addressing exactly this issue. How can we save energy in Clouds, while guaranteeing a high service quality at all times?**

## **Will IT destroy the environment?**

ICT makes up for 2% of the worldwide CO2 production – and rising. This might not seem much, but it is as much as the aviation industry produces, which is made responsible as one important factor of climate change. Even Prince Charles tries hard to reduce the number of flights he takes; should he also give up on using his laptop or smartphone, or deactivate his Twitter account?

## **Energy-efficiency vs Energy-efficiency**

Concerning ICT, an every-day user could save some energy by turning off monitors instead of letting them run on standby. However, developers have many more possibilities: they design energy-efficient algorithms, limit the amount of stored data or data sent over networks, design less energy-hungry routers, or build more energy-efficient hardware components. In my thesis I have explored another way of saving energy: *managing* these (more or less energy-efficient) components in an energy-efficient way without destroying the performance of the overall system!

## **Energy-efficient management of Cloud Computing infrastructures**

So, let's concentrate on Clouds. We won't cover here what a Cloud is or what it is intended to do. What we will look into, though, is how an infrastructure used for Cloud Computing is typically set up, and after that, how we possibly can manage this infrastructure. In Figure 1 we depict three layers of Cloud infrastructures:

1. applications that users want to run in the Cloud;
2. virtual machines (VMs) that these applications are deployed on;
3. and physical machines (PMs) that host the virtual ones.

Users deploy their applications or VMs on the Cloud. They agree on performance guarantees with the Cloud provider (symbolized by the small contracts). Applications or VMs could also

be outsourced to other Clouds as we will see later.

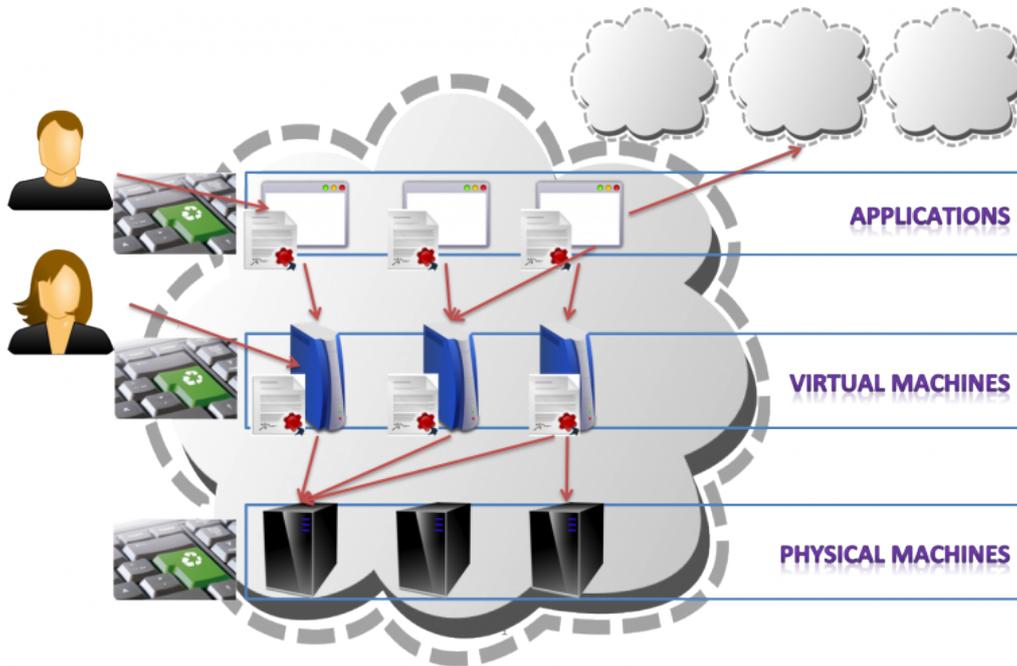


Figure I: A Cloud infrastructure

### Once is not enough

Perhaps you already grasp the challenge here: Finding an efficient placement of the applications on VMs and of the VMs on PMs. Actually, these are all well-studied problems, most of them known as *Bin-packing problems*, *binary integer programming* problems, and similar. Yet, there is a catch (actually two):

1. The challenge we face is not one-dimensional as the figure might suggest. We need to deal with several types of resources such as CPU power, storage, memory, incoming or outgoing bandwidth. It is not sufficient to find a mapping of VMs to PMs optimizing only one resource. Additionally, all these resources behave in a very different way.
2. The demands of the system change quite quickly, e.g., the workload produced by the applications. An application might be a web server that suddenly has a high increase of user interactions due to an unforeseen event (the pope dies or Michael Jackson resurrects), or produces a high amount of data when rendering 3-D images of a surgery, but consumes almost nothing the rest of the day. Thus, a static deployment can turn out quite sub-optimal. Due to this complexity and the quick pace decisions that have to be made, a manual administration of this infrastructure is not possible. Only autonomic management can help!

## Actions, actions, actions

So what can “we” (actually the autonomic manager) do to optimize the energy consumption and resource usage when workloads change so rapidly? Here is a short list:

1. Migrate applications to other VMs
2. Resize VMs in terms of memory, storage, CPU power, bandwidth
3. Migrate VMs to other PMs
4. Power on or off PMs
5. Outsource VMs to other Cloud providers

### **Speculating about resource consumption**

The first thought I had was to resize VMs according to what they really consume. So to not stick to their initial configuration as seen on the Amazon EC2 Cloud for instance, but to adapt their memory, CPU power (cores), storage, etc. to what they really need at the moment. This could mean reducing the allocated memory when not needed such that it is available for other VMs, but to re-allocate enough memory once it is needed to enable a flawless service experience. The assumption to this hypothesis is that when all the VMs have a resource-efficient “size”, it should be easy to pack them on certain PMs so that we can power off the unused ones.

We will see whether the assumption of this speculative approach will hold on my next post. And also what to do with the other possible actions for the autonomic manager. For the more curious ones, you can also read my papers and my thesis meanwhile.