awareness magazine
SELF-AWARENESS IN AUTONOMIC SYSTEMS

10.2417/3201310.005190

Toward a decentralized and resilient autonomic cloud

Philip Mayer and Annabelle Klarl

The integration of cloud computing with voluntary computing and peer-to-peer computing can help clouds to become more distributed and dynamic while maintaining reliability.

The 'cloud' is one of the buzzwords of the current computing season. People store their data in the cloud, share their thoughts through the cloud and use applications hosted in the cloud. Companies outsource their infrastructure and service needs to the cloud to save money and time. The advantage of the cloud is that everything is reliably stored and available on demand at any time and from any location. The data, applications, services and so forth are there whenever and wherever the user needs them, without having to worry about managing anything locally.

Behind all these convenient and lightweight-looking clouds are large data centres with huge server landscapes. Amazon, Microsoft and Google are just some of the big providers of public clouds on which ever more users—both private and corporate—have come to rely. Such cloud systems are offered from one or more centrally coordinated locations, and the servers providing the infrastructure run in well-maintained data centres, under the control of a single entity. All these features make current cloud architectures vulnerable to single-point-of-failure outages.

Yet assuring the reliability of a cloud system is a critical factor for the provisioning companies as well as for the customers that use them. Cloud computing outages may cause severe problems, with service and infrastructure downtimes ranging from a few minutes to many hours. Today, the analysis and repair of these failures is typically performed by human operators, which makes it difficult to predict the time required for re-establishing cloud availability to normal levels.

One proposal for tackling these challenges is to try to integrate some degree of autonomy into clouds, to endow them with a more dynamic and open functionality while at the same time maintaining the key benefits of a cloud, as a reliable and flexible approach for using third-party resources and services. To this end, we can apply the concept of 'autonomic computing', as described by Kephart and Chess in 2003,[1] which addresses the creation of systems that are able to manage themselves, reacting to unforeseen and dynamically evolving situations.

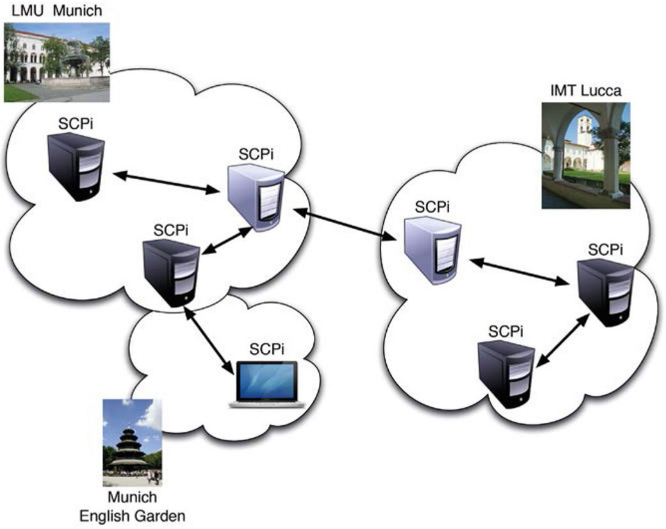

Figure 1. *A schematic view of the cloud nodes, referred to as Science Cloud Platform instances (SCPi), distributed in different locations. IMT: Institute for Advanced Studies. LMU: Ludwig-Maximilians-Universität.*

Applying this autonomic computing paradigm to cloud scenarios results in the vision of an autonomic cloud, i.e., a distributed software system that is able to execute applications in the presence of certain difficulties such as leaving and joining computers, fluctuating usage and applications with different requirements. To realize this vision of an autonomic cloud, in our work we have designed a platform-as-a-service cloud architecture—called the Science Cloud Platform (SCP)—that takes inspiration from and integrates three different computing paradigms: cloud computing, voluntary computing and peer-to-peer computing.[2]

First and foremost, our autonomic SCP is still an implementation of cloud computing, meaning a system for provisioning resources to consumers 'over the net' without them having to install hardware or software themselves. This type of resource provisioning can happen at different levels (infrastructure, platform and service), all of which benefit from the autonomic cloud

Continued on next page

implementation. In our specific case we looked at a platform-as-a-service solution, in which the autonomic cloud software, installed on multiple virtual or non-virtual machines, provides a platform for application execution. The applications running on the SCP may have requirements similar to service-level agreements, specifying the conditions in which they can or prefer to be run. These can include parameters such as CPU speed, available memory or even network proximity (expressed, e.g, in terms of latency) to other applications or nodes.

The second computing paradigm integrated into the SCP concept is voluntary computing. This term usually refers to solutions where individuals offer to contribute a portion of their computing power to a larger computing effort. The classic examples are the @home programs, of which the most famous one is probably SETI@Home,[5] where personal computers are used in the search for extra-terrestrial intelligence. In our autonomic SCP, we envisage resources (that is, computers) being provided voluntarily by individual people, but also by entities such as universities: see Figure 1. Such contributed cloud nodes can be added or removed at any time, as their owners see fit. This means that nodes may come and go without warning, and that their load may change due to causes external to the cloud. A voluntary cloud would therefore include nodes with vastly different hardware, e.g., in terms of CPU speed, available memory or the presence of specialized hardware such as graphics processing chips.

Finally, the third computing paradigm incorporated into our autonomic SCP concept is peer-to-peer computing. First popularized in the controversial area of file sharing, the basic idea of peer-to-peer computing is that of an organization without a centralized structure. There is no single node in the network on which the functionality of the overall system depends. Instead, a decentralized communication approach is used that is ideally able to maintain stability even while nodes are continually joining and leaving, and thus offers no single point of failure or single point of attack. Our autonomic SCP is likewise based on this idea. There is no centralized component in the cloud, and so the nodes must use some protocol to agree, in a decentralized manner, on what to execute and where.

Thus, our overall vision of the autonomic cloud is that of a voluntary, peer-to-peer-based platform-as-a-service solution. This goal can be pursued by introducing a sense of self-awareness to clouds to enable them to autonomously react to changes in the infrastructure. The autonomic nodes must be (self-)aware of changes in load (arising either from cloud applications or from local applications external to the cloud) and in the network structure (i.e., nodes coming and going), which calls for self-healing properties. In the case of nodes dropping out of the system, the system must also implement redundant data storage to prevent data loss. Finally, executing applications in such an environment requires a fail-over solution, meaning self-adaptation of the cloud to provide what we may call application execution resilience.

In summary, our work explores the idea of an autonomic cloud system that, when implemented in the manner described, will be able to deploy and run user-defined applications on a peer-to-peer connected web of voluntarily provided machines. It will be able to adapt to different load scenarios and numbers of machines while still achieving the central cloud-computing goal of keeping data safe and applications running. We plan to continue this work by further researching various aspects, such as better investigating self-adaptation performance in the cloud, carrying out large-scale tests and exploring alternative implementation models.

This work was carried out as part of the EU project Autonomic Service-Component Ensembles (ASCENS).[4] *The authors wish to thank all the partners for their ideas and collaboration on the autonomic cloud vision.*

Author Information

Philip Mayer and Annabelle Klarl
Institute for Informatics
Ludwig-Maximilians-Universität
Munich, Germany

References

1. J. Kephart and D. Chess, *The vision of autonomic computing*, **IEEE Computer 36** (1), pp. 41–50, 2003.
2. P. Mayer, A. Klarl, R. Hennicker, M. Puviani, F. Tiezzi, R. Pugliese, J. Keznikl, and T. Bures, *The autonomic cloud: a vision of voluntary, peer-2-peer cloud computing*, **Proc. 3rd Wrkshp. Challenges Achieving Self-Awareness Autonom. Syst.**, pp. 1–6, 2013.
3. F. Zambonelli, N. Bicocchi, G. Cabri, L. Leonardi, and M. Puviani, *On self-adaptation, self-expression and self-awareness in autonomic service component ensembles*, **Proc. 5th IEEE Conf. Self-Adapt. Self-Organiz. Syst. Wrkshp. (SASOW)**, pp. 108–113, 2011.
4. M. Wirsing, M. Hölzl, M. Tribastone, and F. Zambonelli, *ASCENS: engineering autonomic service-component ensembles*, in B. Beckert, F. Damiani, M. Bonsangue, and F. de Boer (eds.), **Formal Methods for Components and Objects, Lect. Notes Comput. Sci. 7542**, pp. 1–24, Springer, 2013.
5. E. Korpela, D. Werthimer, D. Anderson, J. Cobb, and M. Lebofsky, *SETI@home—massively distributed computing for SETI*, **Comput. Sci. Eng. 3** (1), pp. 78–83, 2001.