Cloud Computing

Cloud Computing, Spring 2024

Universiteit Leiden

Cloud Computing

Universiteit
Leiden

Cloud Computing

- Lecturer: Dr. Kristian F. D. Rietveld
  - TA: Weikang Weng MSc
- Full semester course, 6 EC
  - Designed for and mandatory for “Advanced Computing and Systems” (ACS) track.
  - Limited capacity for students from other tracks.
  - Selection procedure for remaining places.
  - This will be an advanced **computer systems** course, therefore participants are expected to have taken *computer architecture* and *operating systems* courses on a BSc level.
- Brightspace course: all information can be found there.
  - All students that participate with the first assignment & paper selection will receive a final grade, also if the remainder of the course is not finished.

Course plan

- Study the research area of Cloud Computing together.
- Planned structure:
  - ~6 broad **introductory lectures** by the lecturer.
  - **Paper presentations** by all course participants.
    - Recently published paper (last ~3 years, 8+ pages). Lecturer to provide a list of papers to choose from.
  - **Home work** and **Lab assignments** will run along lectures and presentations.
  - Final lecture: wrap-up / closing / what did we learn?
  - **Written exam** in June.

Examination

- The final grade consists of three parts:
  - Lab assignments (40%)
  - Paper presentation (25%)
    - Form your own opinion of the content and scientific contribution of the paper; in fact a mini review.
    - Peer review: your presentation will be graded by the lecturer and two randomly picked participants. You are required to participate in peer review.
  - Written exam with open & essay questions (35%)
- All components must be sufficient (>= 5.5) to pass the course.
- Homework is mandatory but don't count against the final grade.

Lab Assignments

- Teams of 2 are allowed.
- Linux & programming skills required.
- Two lab assignments are planned:
  - 1. Small, automatically scaling web application on top of Linux container technology and virtual networking. **40%**
  - 2. Write a Cloud-like application on top of typical components used in Cloud software. Scaling & fault tolerance experiments. Goal: become familiar with Cloud technologies. **60%**

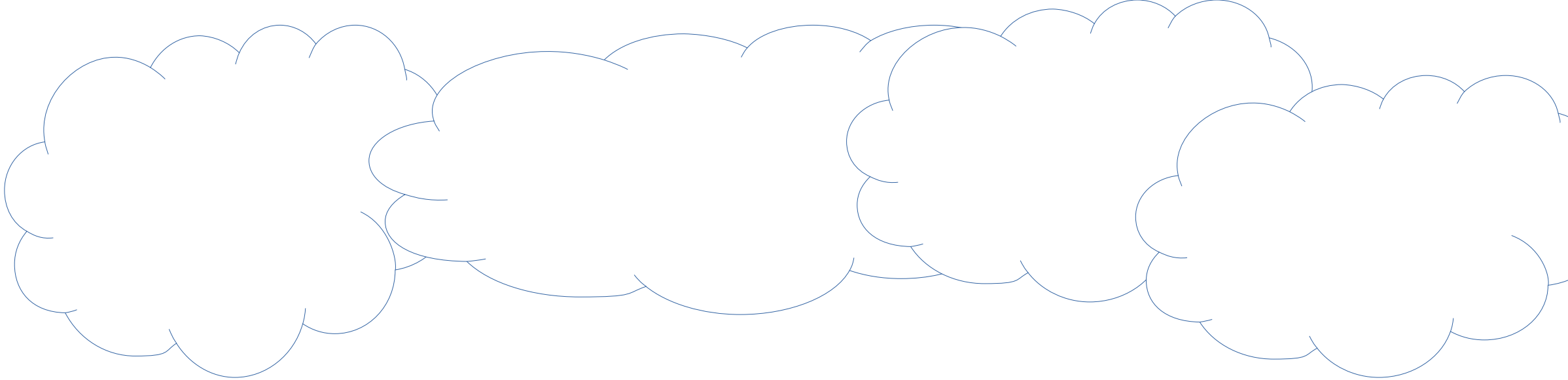
Textbook

- The course does not have a prescribed textbook; the lecturer couldn't find something suitable.
- Therefore, we will read academic literature.
- A reading list of papers used as source for these slides will be made available on BrightSpace.

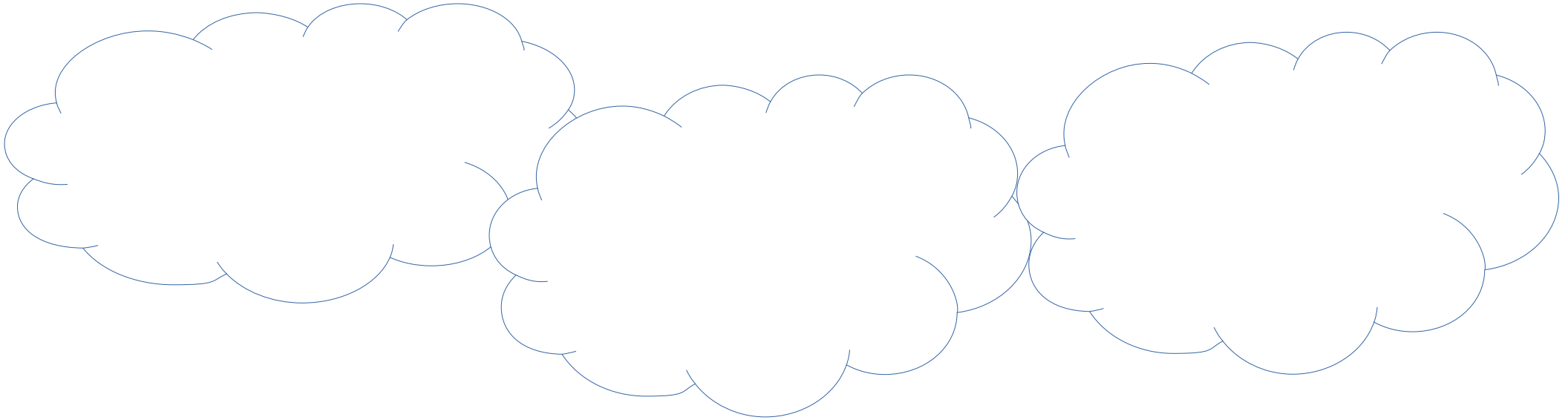
Course contents

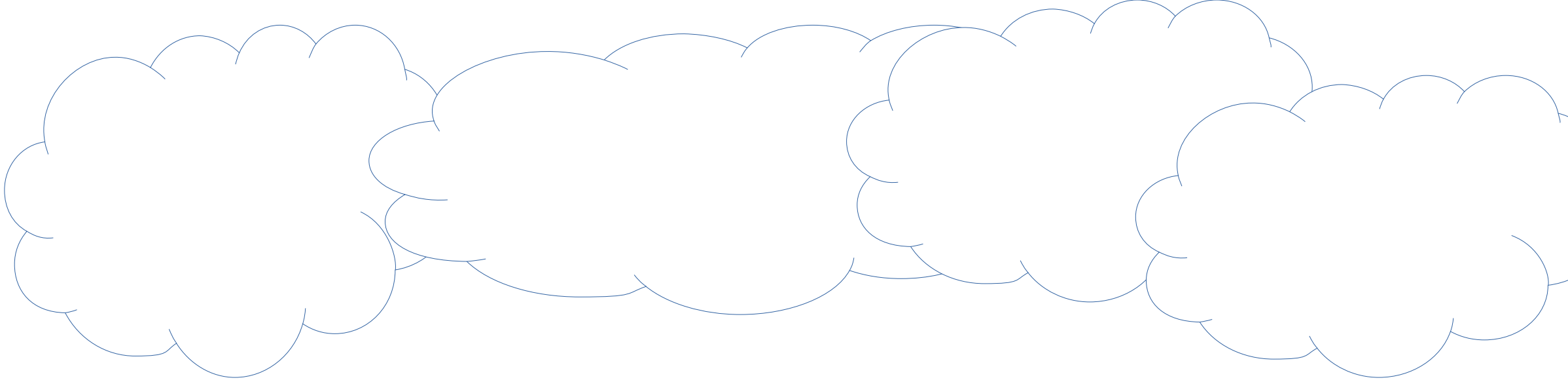
- What will we be discussing?

What is Cloud Computing anyway?



What is Cloud Computing anyway?





What is Cloud Computing anyway?

Your ideas?

What is Cloud Computing?

- This is actually a good question, and several papers have been written on this topic!
  - E.g. L. Vaquero et al. (2009): *A Break in the Clouds: Towards a Cloud Definition*.
- An MIT technology review article suggests that
  - Cloud computing was coined as marketing term in a 1996 business proposal.
  - In 2006 the term started to be used by big companies (Google, Amazon) to describe SaaS architectures.
  - Source: https://www.technologyreview.com/s/425970/who-coined-cloud-computing/

What is Cloud Computing (2)

- What do different papers have to say?

- L. Vaquero et al. (2009): overall picture confusing; but boils down to a large collection of virtualized resources that can be dynamically scaled and are billed through pay-per-use model.
- G. Pallis (2010): new multidisciplinary research field, consisting of several trends coming together.
- M. Armburst et al. (2010): the combination of SaaS and utility computing.

NIST definition

- Published in September 2011.
  - Source: https://csrc.nist.gov/publications/detail/sp/800-145/final
- Five essential characteristics:
  - *On-demand self-service*: customers can provision compute time and data storage without requiring intervention of a system administrator at the service provider.
  - *Broad network access*: widely accessible over the network using standard protocols.
  - (see next slide)

NIST definition (2)

- Five essential characteristics:
  - *Resource pooling*: with the same set of resources multiple customers are served; resources can be dynamically reassigned. Customer has no control over exact location of resource (apart from specifying region).
  - *Rapid elasticity*: automatic acquisition and release of resources to enable up- and downscaling. Available capacity appears unlimited to the customer.
  - *Measured service*: all use of resources is metered. Can be used for monitoring and billing.

Novelty

- How novel is Cloud Computing?
- There appears to be consensus that Cloud Computing is in fact the combination of a number of existing technologies.
- The individual technologies are not new:
  - Virtualization has been around for a long time.
  - Utility computing, the idea of paying another company as you go to run your computations, was already thought about in the '60s and '70s.
  - Service metering is common practice for computational clusters.
  - Broad network access is also not new, although it is true that this has improved significantly in the last decade.

What about Grid Computing?

- The idea of grid computing was to combine multiple cluster computers (administered by different organizations) into a “single” large cluster computer.
  - Not really *on-demand self-service* and *rapid elasticity*, but rather classic HPC job scheduling.
- Special grid middleware was developed to achieve this.
- The different clusters are typically heterogeneous and located at different geographical locations.
- Original HPC grids were usually private.

- The largest grid is the one built for CERN LHC. It comprises 170 computing centers in 42 countries (http://wlcg-public.web.cern.ch/).

NIST definition (3)

- Three service models:
  - Software as a Service (SaaS)
  - Platform as a Service (PaaS)
  - Infrastructure as a Service (IaaS)

- Of course many more have been devised. XaaS.

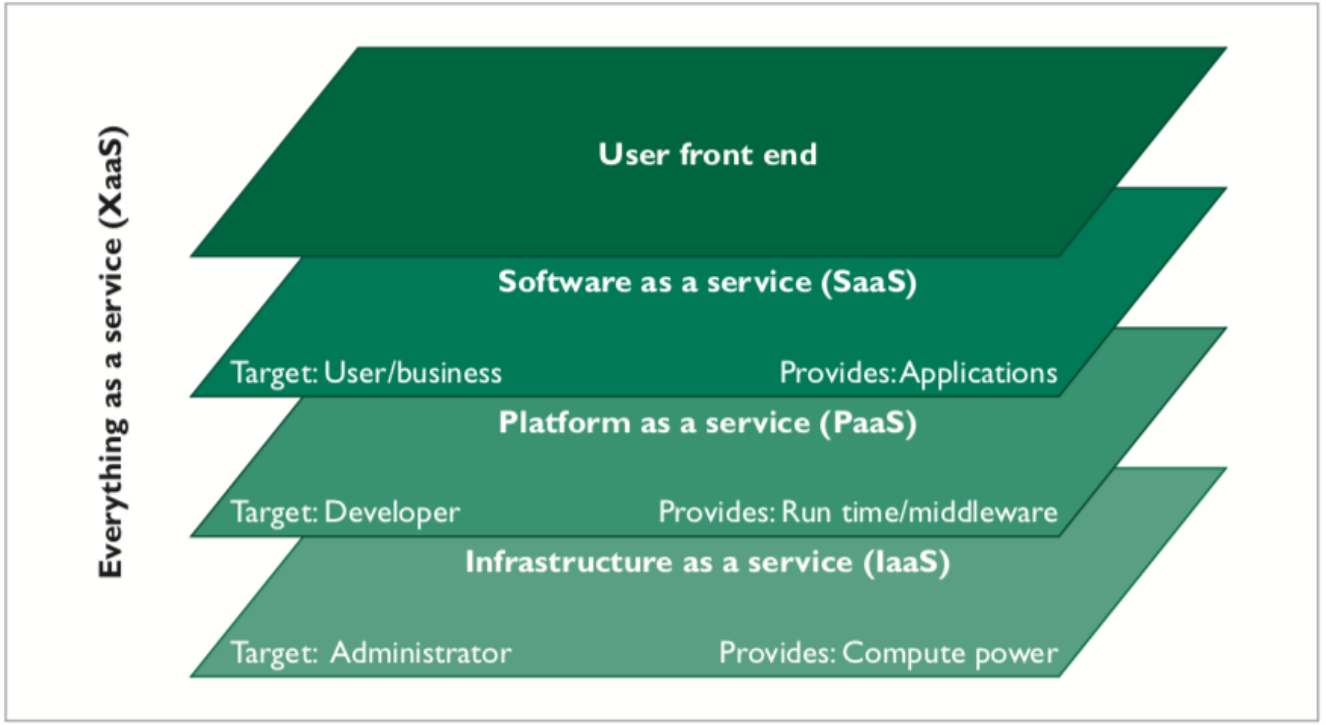


Figure 1. A general layered architecture of cloud infrastructures. Cloud computing uses IT infrastructure as a service. Its architecture defines three distinct layers from computing resources to end-user applications.

Image source: G. Pallis (2010): *Cloud Computing, The New Frontier of Internet Computing.* IEEE Internet Computing.

SaaS

- Applications running on a provider's cloud infrastructure, accessible through a “thin client” interface (for example web browser, smartphone App).
- Examples: web-based e-mail, web-based document processing, social media platforms, online bookshops.

PaaS

- Deploy applications onto cloud infrastructure, written using languages/libraries/services supported by the cloud provider.
- This is an abstract deployment, there's no control of underlying hardware resources.
- Examples: Google App Engine
  - Ability to develop web applications in one of the supported languages (Python, Java, Go, PHP, Node.js)
  - Access to several frameworks
  - Automatic upscaling on demand
  - Code can only be called from a HTTP request

IaaS

- “Rent” computing resources: CPU time, storage, networking. Arbitrary software can be run on these resources (cf. PaaS!).
- Think about provisioning virtual machines, block and object storage.
- Examples: Amazon EC2, Google Compute Engine, RackSpace, Linode and many other VPS (virtual private service) providers.

Addition: Serverless Computing

- FaaS: Functions as a Service
- The idea is to write functions that are hosted “in the cloud”.
  - No need to think about servers and infrastructure.
  - Function is invoked by an app running on a terminal (e.g. smartphone) and then executed.
- You pay per function invocation (or accumulated function running time).
  - When no code is executing, there is no charge (cf. IaaS)
- Only functions are hosted in the cloud, not an entire web application (cf. PaaS).

NIST definition (4)

- Four deployment models:
  - *Private Cloud*: cloud infrastructure private to a single organization.
  - *Public Cloud*: infrastructure open for use by general public.
  - *Community Cloud*: exclusive use by a specific community of users.
  - *Hybrid Cloud*: infrastructure that is any combination of the above.
    - Example: surge computing: we run everything on a private cloud, but in case of peak demand also resources from a public cloud are used.

Addition: multi-cloud

- Use resources from multiple cloud providers, because you might not want to rely on a single provider.
- Interoperability challenge:
  - Underlying virtualization techniques might be different.
  - Different APIs for managing resources.
  - Different SLA / cost.
  - Different authentication techniques.
- Concisely summarized and described in:
  R. Ranjan (2014): *The Cloud Interoperability Challenge*. IEEE Cloud Computing.

Addition: Federated Clouds

- Multiple organizations pooling their cloud infrastructure together, to create a single larger infrastructure.
- Individual cloud infrastructures may be heterogeneous, middleware required to make this work (and to solve interoperability challenges).

- Other examples of federated technologies are SMTP (e-mail) and XMPP (Jabber messaging).

Business Aspects (Cloud Economics)

- Several papers tout the advantages of cloud computing for businesses:
  - Start-up companies don't need a lot of capital up front to build their own infrastructure.
  - Much easier to deal with peak demand:
    - It is no longer required to have this on premises, you can simply rent infrastructure when necessary.
    - Also much cheaper: idle machines on premises are expensive, in the case of cloud computing you pay as you use.
  - For smaller companies: no need to have in-house system administration, so they can focus on their product.

Topics to be covered

- Technical fundamentals of cloud infrastructure
- Resource management, availability, elasticity
- Cloud programming models and tools
- Emerging topics in cloud computing

(We will explicitly not discuss economics, management, DevOps software engineering)

Next week

- We discuss the technical foundations of Cloud Computing.
  - Virtual machines, Hypervisors, trap-and-emulate, CPU extensions, nested page tables, …

- Information on selection procedure will follow on Brightspace.

Universiteit Leiden