

Cloud Computing

Walfredo Cirne

Agenda

- What is the cloud?
 - Motivation
 - Cloud provisioning
 - SLO variation
 - IaaS, PaaS, SaaS
- Using the cloud
 - Dependency and locking
 - Pets x Cattle
- Trust & security

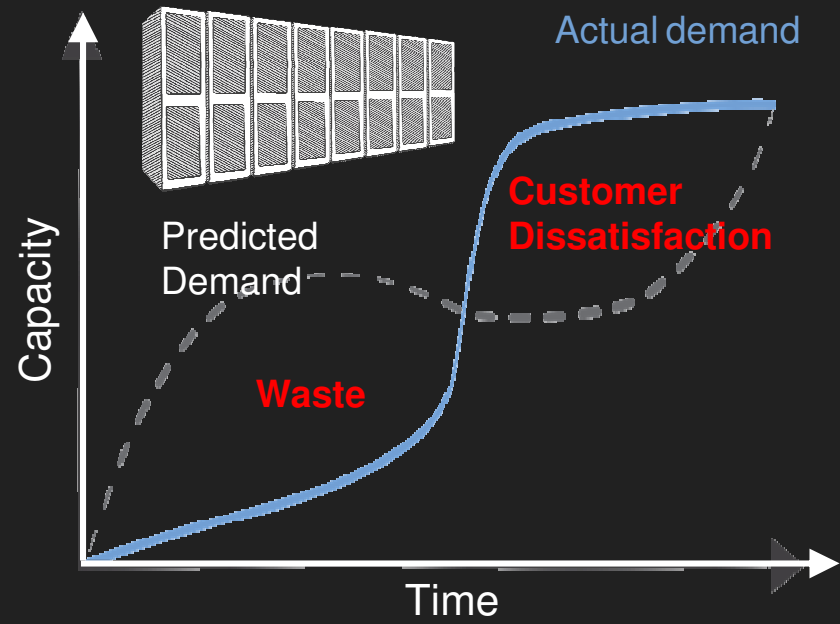
What is cloud computing?

- Computing today is distributed ~ client/server
- Anytime, anywhere access to your information from any device
- If your computer isn't connected to the Internet, it is "broken"
- Cloud Computing = You rent the servers, which are somewhere else in the world

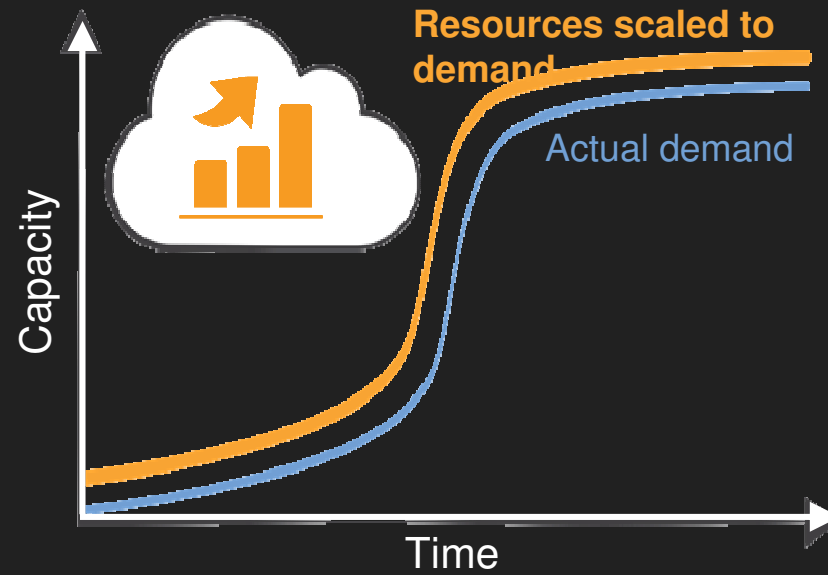
Motivation

- Economy of scale
- It is cheaper (per computer) to provision 1000 computers than 1
- Even better economics when peak consumption is not correlated
- The same idea of power generation
 - Cloud computing ~ power grid
 - Your own computer ~ generator

Rigid local computing



Cloud Elastic Computing



Economy of Scales

- Economy in capital expenses
- Economy in operational costs
- Larger and more heterogenous load is smoother

Cloud Provisioning

- As a provider: How much machines to do keep around such that you have VMs to sell when consumers show up?
- As a client: How confident are you that your provider will have VMs available when you need them?

SLO variation

- It is important to understand which SLO (Service Level Objectives) the provider promises ... as this should affect price
- Different applications can survive at different SLO levels
 - Amazon's on-demand x spot-market VMs
 - Google's normal x preemptable VMs
 - The commercial version of SETI@home?

Why lower grade SLOs?

- Because it is cheaper
- The question really is: Can I actually use them?
- Well, it depends of you application

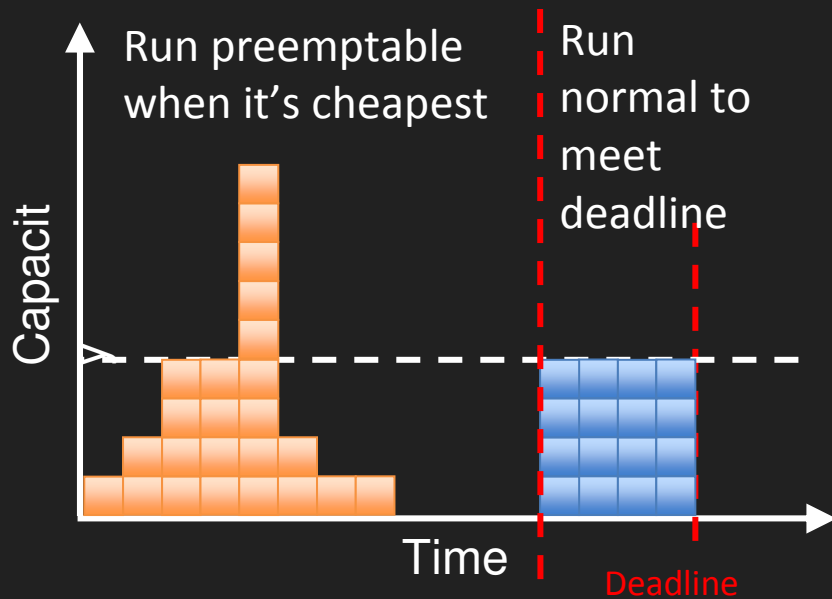
- Running a web server? Really, really hard
- Running a long, experimental simulation? Fine
- Running a “production batch”? How does its value vary over time? Does it have deadlines?

Examples?

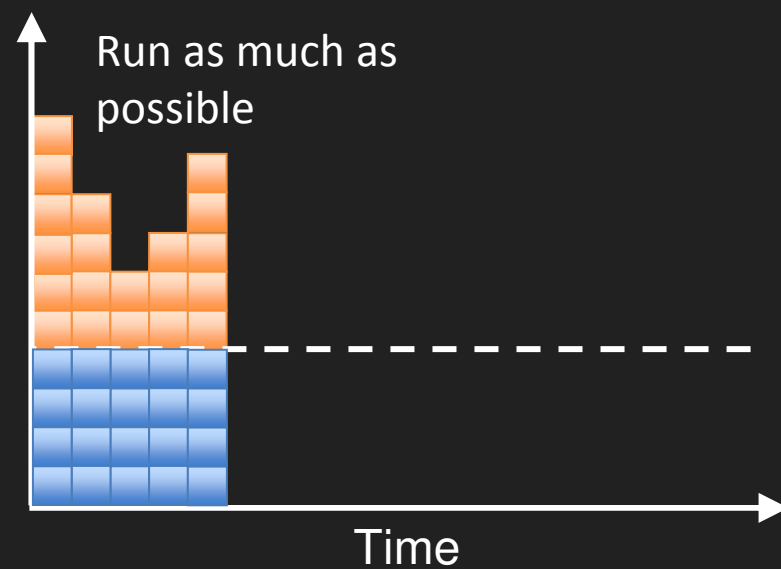
- Weekend regression tests
 - Take 18 hours to run
 - Fine if done Monday at 8:00
 - Possible strategy: Use preemptible VMs to make progress until Sunday 14:00, and then switch to normal VMs
- Financial analysis
 - Makes sense in normal VMs
 - Possible strategy: Use preemptible VMs to make it faster/cheaper

Different Strategies to use Preemptable VMs

Case 1:
Value of result is steady until deadline



Case 2:
Value of result quickly diminishes



Applications amenable for preemptable VMs

- Fault tolerant (e.g. checkpointing)
- Elastic/scaleble

SPI model - SaaS, PaaS, IaaS

- Where in the stack do you buy cloud?
- IaaS - Infrastructure as a Service
You buy compute and storage
- PaaS - Platform as a Service
You buy a service for developers to use, e.g. a database
- SaaS - Software as a Service
You buy access to something designed for end users
- The cloud “ecosystem”

Dependency & Locking

- Cloud computing = Normal computing, for dependency & locking
- The higher in the stack you buy, the more locked you typically are
 - IaaS is easier to replace than SaaS
- The higher in the stack you buy, the more value you get
 - SaaS is what your users want
- The cloud “ecosystem”
- Standards help to reduce dependency & locking
 - Kubernetes is a really interesting open-source, Google-backed cluster manager

Pets x Cattle

- How do you care for your virtual machines?
 - Are they “pets”? Do they have names? Do you take them to the veterinarian?
 - Are they “cattle”? Do they have numbers? Do you just replace them when they get sick?
- Configuration is one of the biggest open problems in computing

Trust & Security

- Are you really putting your data in someone's computer?
 - Is this even legal?
- Are you confident your computer is as well defended as Google/Amazon/Microsoft/IBM?
 - On the other hand, they are bigger targets
- You don't need one size fits all
 - You can combine cloud with local computing

Conclusion

- Cloud is quite exciting
- It makes computing cheaper for everybody
- It provides a better vantage point to improve compute utilization

`brigado