

Advanced Software Engineering

Master Degree in Computer Science

Master Degree in Computer Science and Networking

Antonio Brogi

Department of Computer Science

University of Pisa

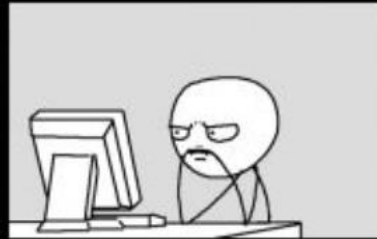
A man with a beard is shown from the chest up, wearing a black t-shirt. The t-shirt has a green text graphic that reads: "A software engineer is a device that turns coffee into code." The text is arranged in five lines, centered on the chest. The background is a plain, light gray.

A software
engineer is a
device that
turns coffee
into code.

Software Engineers



What Society Believes I Do



What My Wife Believes I Do



What My Mom Believes I Do



What My Boss Believes I Do



What I Believe I Do



What I Really Do

Service-based software engineering

Service-based software engineering
Service-based economy

Everything as a service

- Economy trend: from “*goods*” to “*services*”



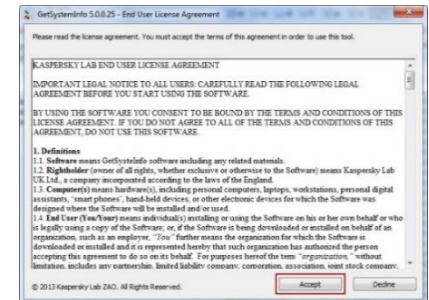
Service contracts

- Customers do not (want to) know how the service they use is actually implemented



- Customers (should) choose whether to use a service or not on the basis of the *contract* that is (or should) be exposed by the service provider

... but ... do customers
read contracts at all?



Service-based software engineering

Service-based economy

QoS & SLAs

QoS matters, though



vs.



QoS matters, though



To store **all the data** of my school, I have chosen the storage service offered by the new local software company *StoreForLess*.

We will save **50 USD** per year if compared for instance with using Amazon S3!

Service Level Agreements



Effective Date: June 1, 2014



This Amazon S3 Service Level Agreement ("SLA") is a policy governing the use of Amazon Simple Storage Service ("Amazon S3") under the terms of the Amazon Web Services Customer Agreement (the "AWS Agreement") between Amazon Web Services, Inc. and its affiliates ("AWS", "us" or "we") and users of AWS' services ("you"). This SLA applies separately to each account using Amazon S3. Unless otherwise provided herein, this SLA is subject to the terms of the AWS Agreement and capitalized terms will have the meaning specified in the AWS Agreement. We reserve the right to change the terms of this SLA in accordance with the AWS Agreement.

Service Commitment

AWS will use commercially reasonable efforts to make Amazon S3 available with a Monthly Uptime Percentage (defined below) of at least 99.9% during any monthly billing cycle (the "Service Commitment"). In the event Amazon S3 does not meet the Service Commitment, you will be eligible to receive a Service Credit as described below.

Definitions

- "Error Rate" means: (i) the total number of internal server errors returned by Amazon S3 as error status "InternalError" or "ServiceUnavailable" divided by (ii) the total number of requests during that five minute period. We will calculate the Error Rate for each Amazon S3 account as a percentage for each five minute period in the monthly billing cycle. The calculation of the number of internal server errors will not include errors that arise directly or indirectly as a result of any of the Amazon S3 SLA Exclusions (as defined below).
- "Monthly Uptime Percentage" is calculated by subtracting from 100% the average of the Error Rates from each five minute period in the monthly billing cycle.
- A "Service Credit" is a dollar credit, calculated as set forth below, that we may credit back to an eligible Amazon S3 account.

Service Credits

Service Credits are calculated as a percentage of the total charges paid by you for Amazon S3 for the billing cycle in which the error occurred in accordance with the schedule below.

Monthly Uptime Percentage

Service Credit Percentage

Equal to or greater than 99% but less than 99.9%	10%
Less than 99%	25%

Q: How much *Service Credit Percentage* will you get if S3 was not available for two whole days (April 2nd and 3rd)?

Q: How many "InternalError" or "ServiceUnavailable" errors did you get? And When?

A: I tried to access S3 at 9am, 11am, 1pm, and 3pm both days, getting a "ServiceUnavailable" error every time

- 8 "five minute periods" with Error Rate = 100%
- Remaining $(12 \times 24 \times 30) - 8$ five minute periods with Error Rate = 0%
- "monthly Uptime Percentage" = $100\% - (8 / (12 \times 24 \times 30) \times 100\%) = 99,90740741\%$

Microsoft Services Agreement

Effective July 31, 2014

6. Services disruptions and backup

We strive to keep the Services up and running; however, all online services suffer occasional disruptions and outages. **You should regularly back up the Content that you store on the Services.** Having a regular backup plan and following it can help you prevent loss of your Content. We make an effort to provide detailed service status on the Service status website (<http://status.live.com>).

11. NO WARRANTIES

MICROSOFT, AND OUR AFFILIATES, RESELLERS, DISTRIBUTORS, AND VENDORS, MAKE NO WARRANTIES, EXPRESS OR IMPLIED, GUARANTEES OR CONDITIONS WITH RESPECT TO YOUR USE OF THE SERVICES. YOU UNDERSTAND THAT USE OF THE SERVICES IS AT YOUR OWN RISK AND THAT WE PROVIDE THE SERVICES ON AN “AS IS” BASIS “WITH ALL FAULTS” AND “AS AVAILABLE.” **MICROSOFT DOESN'T GUARANTEE THE ACCURACY OR TIMELINESS OF INFORMATION AVAILABLE FROM THE SERVICES.** TO THE EXTENT PERMITTED UNDER YOUR LOCAL LAW, WE EXCLUDE ANY IMPLIED WARRANTIES, INCLUDING FOR MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, WORKMANLIKE EFFORT, AND NON-INFRINGEMENT. YOU MAY HAVE CERTAIN RIGHTS UNDER YOUR LOCAL LAW. NOTHING IN THIS AGREEMENT IS INTENDED TO AFFECT THOSE RIGHTS, IF THEY ARE APPLICABLE. YOU ACKNOWLEDGE THAT COMPUTER AND TELECOMMUNICATIONS SYSTEMS AREN'T FAULT-FREE AND OCCASIONAL PERIODS OF DOWNTIME OCCUR. **WE DON'T GUARANTEE THE SERVICES WILL BE UNINTERRUPTED, TIMELY, SECURE, OR ERROR-FREE OR THAT CONTENT LOSS WON'T OCCUR.**



For content that is covered by intellectual property rights, like photos and videos (IP content), you specifically give us the following permission, subject to your privacy and application settings: you grant us a non-exclusive, transferable, sub-licensable, royalty-free, worldwide license to use any IP content that you post on or in connection with Facebook (IP License). This IP License ends when you delete your IP content or your account unless your content has been shared with others, and they have not deleted it.

Service-based software engineering

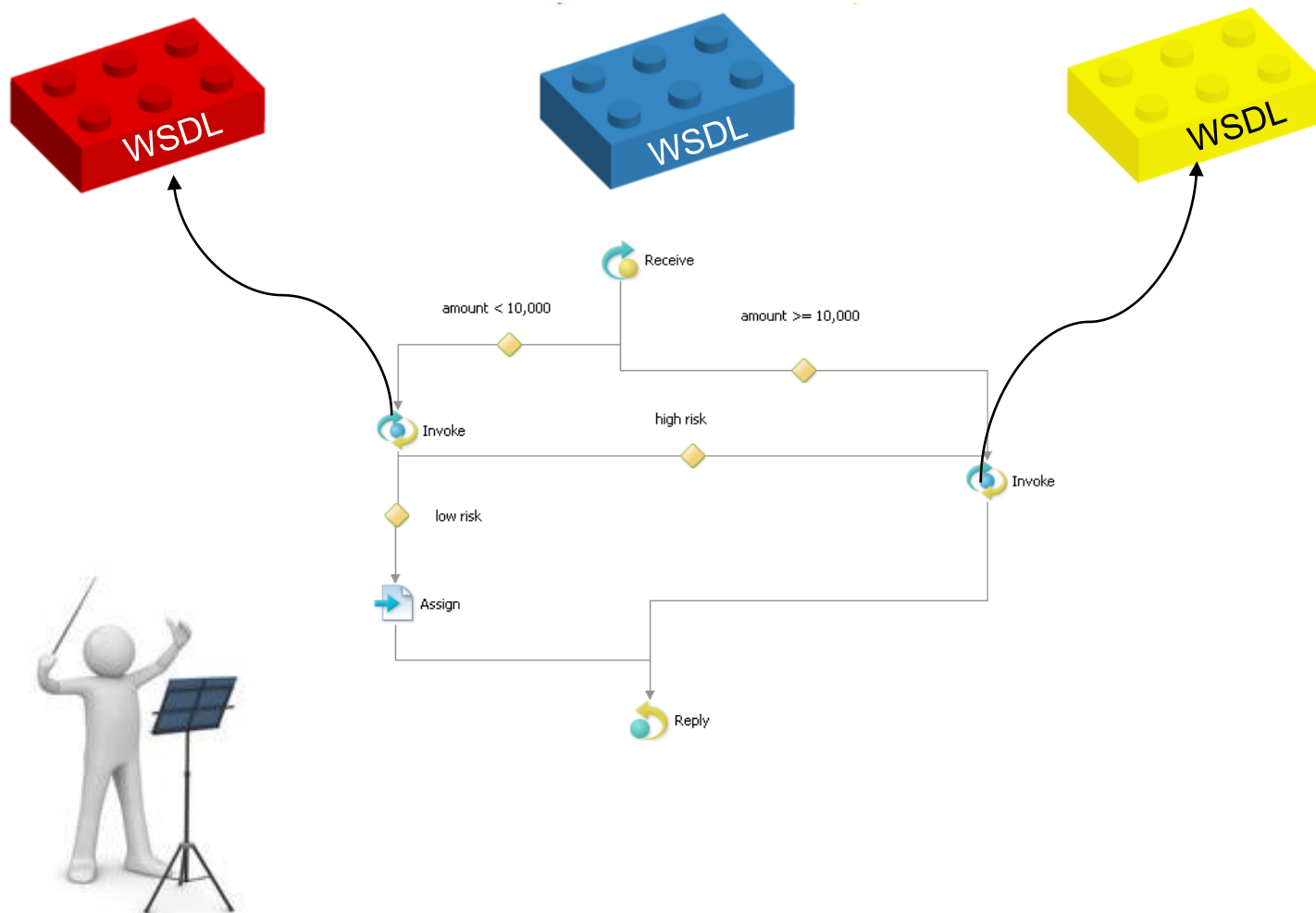
Service-based economy

QoS & SLAs

Service-oriented computing

Service-oriented computing

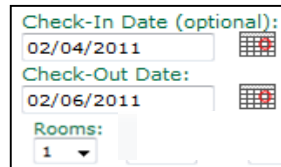
Idea: Use software services to develop distributed applications rapidly and at low cost



Web Services

(1) Web sites vs. Web services

- interfaces



A screenshot of a web form for hotel booking. It contains three input fields: 'Check-In Date (optional):' with the value '02/04/2011', 'Check-Out Date:' with the value '02/06/2011', and 'Rooms:' with a dropdown menu showing '1'. Each date field has a small calendar icon to its right.

```
<operation name = "CheckAvailability">  
  < input message = "CheckInDate" />  
  < input message = "CheckOutDate" />  
  < input message = "NRooms" />  
  < output message = "Result" />  
</operation>
```

- customers



- interaction



Web Services

(1) Web sites vs. Web services

(2) Evolution of software development



object-oriented
programming



component-based
software engineering

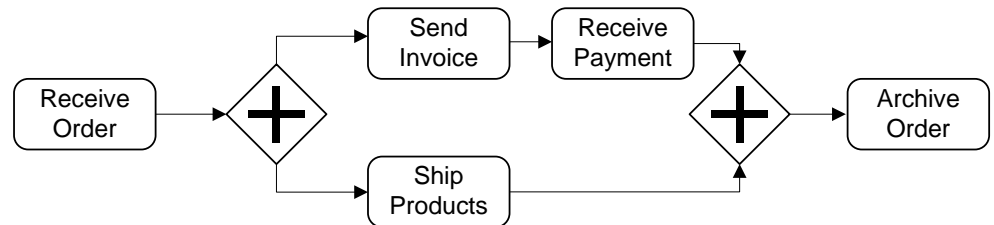


service-oriented
computing

Web Services

- (1) Web sites vs. Web services
- (2) Evolution of software development
- (3) Business process management view

“When we receive a new order, an invoice should be sent to the customer. The order should be archived only after receiving the payment. The requested products must be shipped to the customers.”



Service-oriented computing



- The QoS of a service-based app does depend on the QoS of the services it invokes
 - *What is the QoS of app A invoking S1 and S2?*
 - *What happens if S1 is replaced by S1' ?*
- QoS prediction difficult because of
 - *different results of service invocations*
 - *non-determinism in composition workflow*
 - *correlations & dependencies among activities*

Service Level Agreements

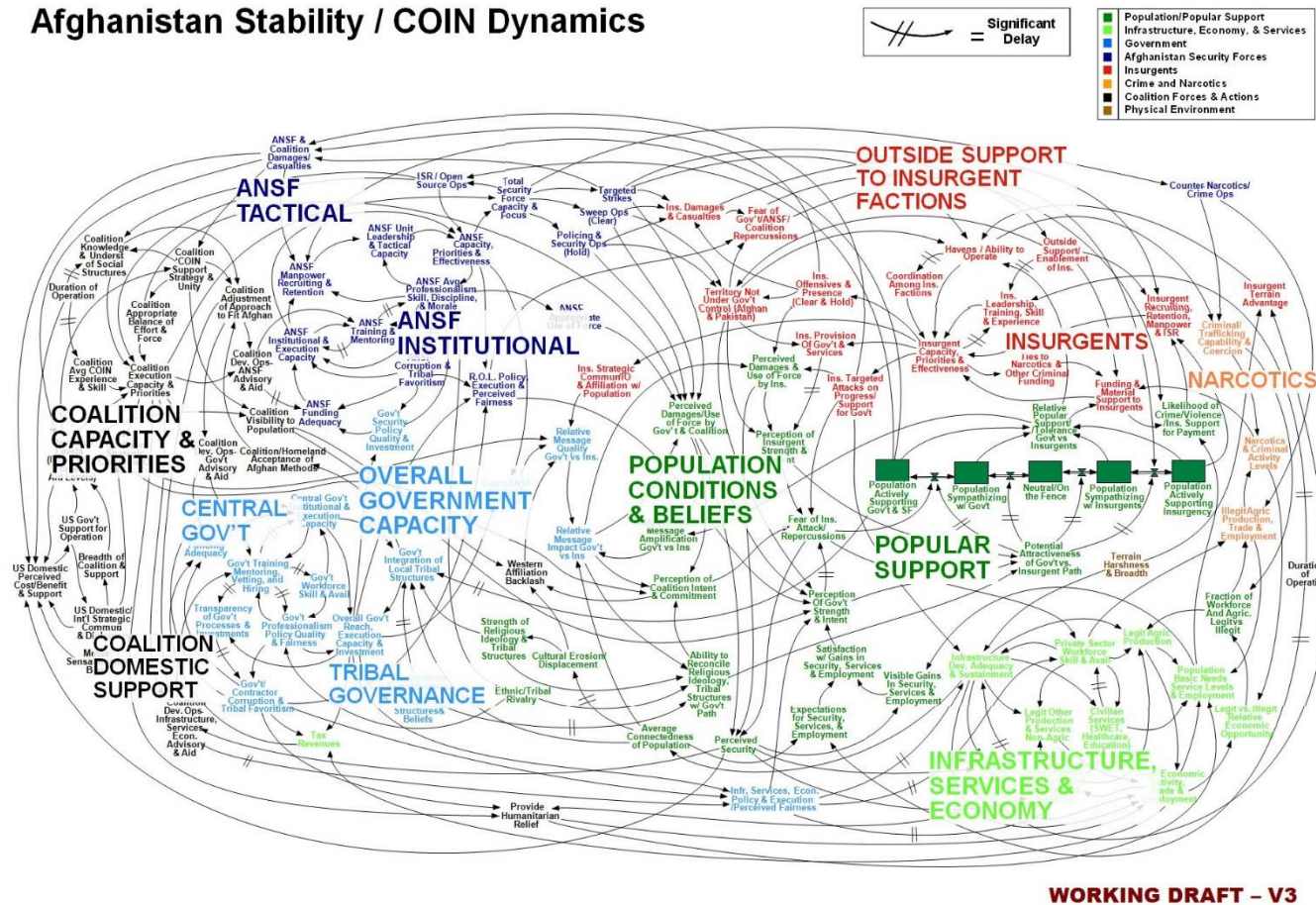


Which reliability can we guarantee for our app?



Service composition

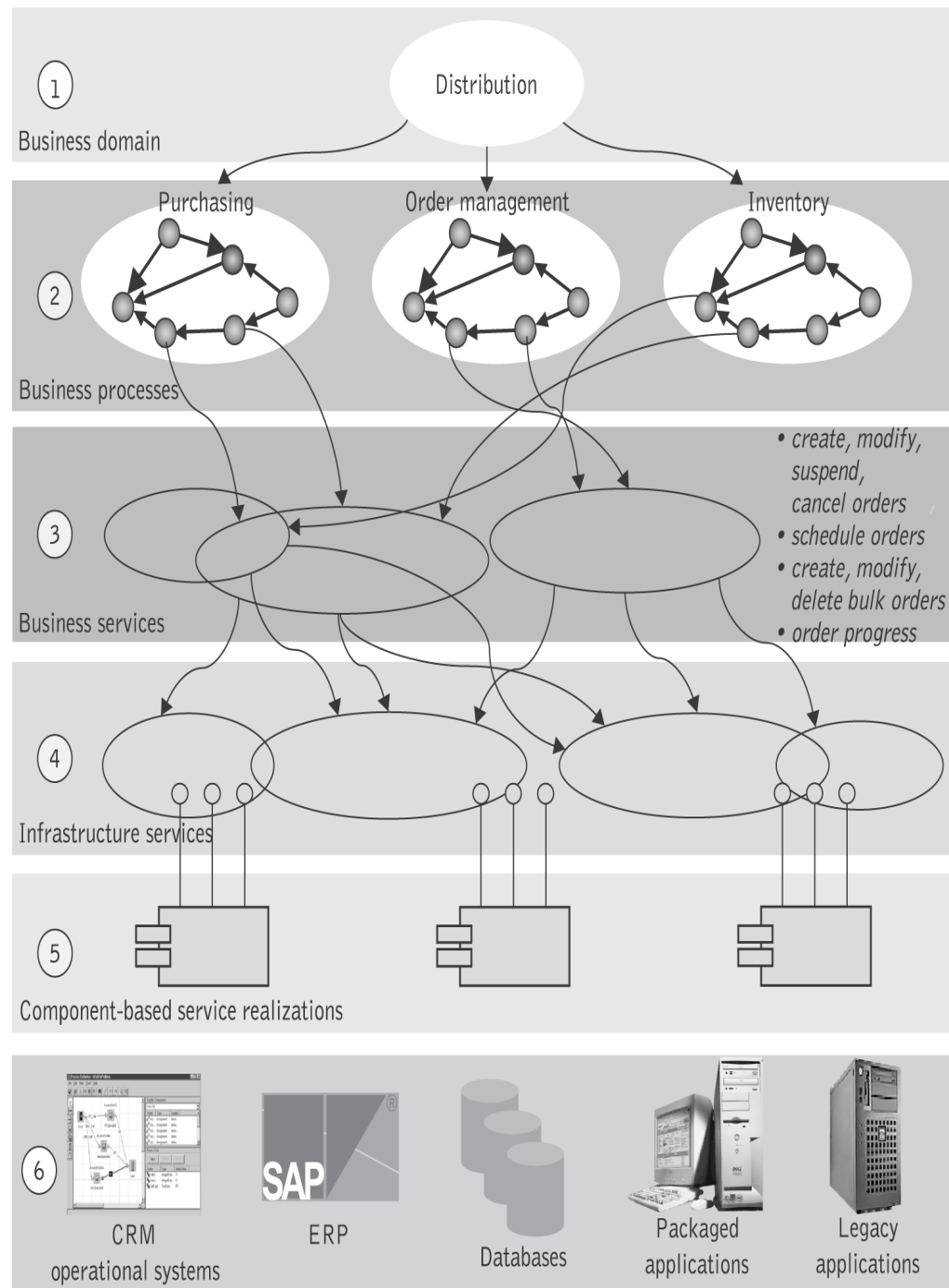
Afghanistan Stability / COIN Dynamics



Correctness of service compositions?

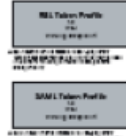
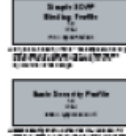


SOA



WS-* standards

Interoperability Issues



Standards Bodies

ISO/IEC International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) jointly develop and publish international standards.

IEEE Institute of Electrical and Electronics Engineers develops and publishes standards for electrical, electronic, and computer engineering.

ITU-T International Telecommunication Union - Telecommunication Standardization Sector develops and publishes standards for information and communication technologies.

W3C World Wide Web Consortium develops and publishes standards for the World Wide Web.

OASIS Organization for the Advancement of Structured Information Standards develops and publishes standards for various information technologies.

ECMA European Committee for Standardization develops and publishes standards for various information technologies.

ISO/IEC JTC1 SC30 Joint Technical Committee 1, Subcommittee 30, focuses on standards for Web Services.

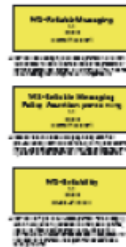
Business Process Specifications



Metadata Specifications



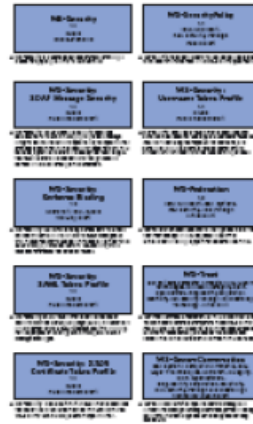
Reliability Specifications



Management Specifications



Security Specifications



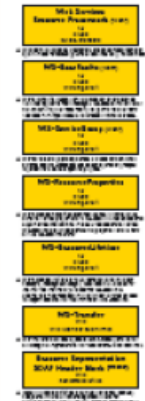
Transaction Specifications



Presentation Specifications



Resource Specifications



Messaging Specifications



SOAP



XML Specifications

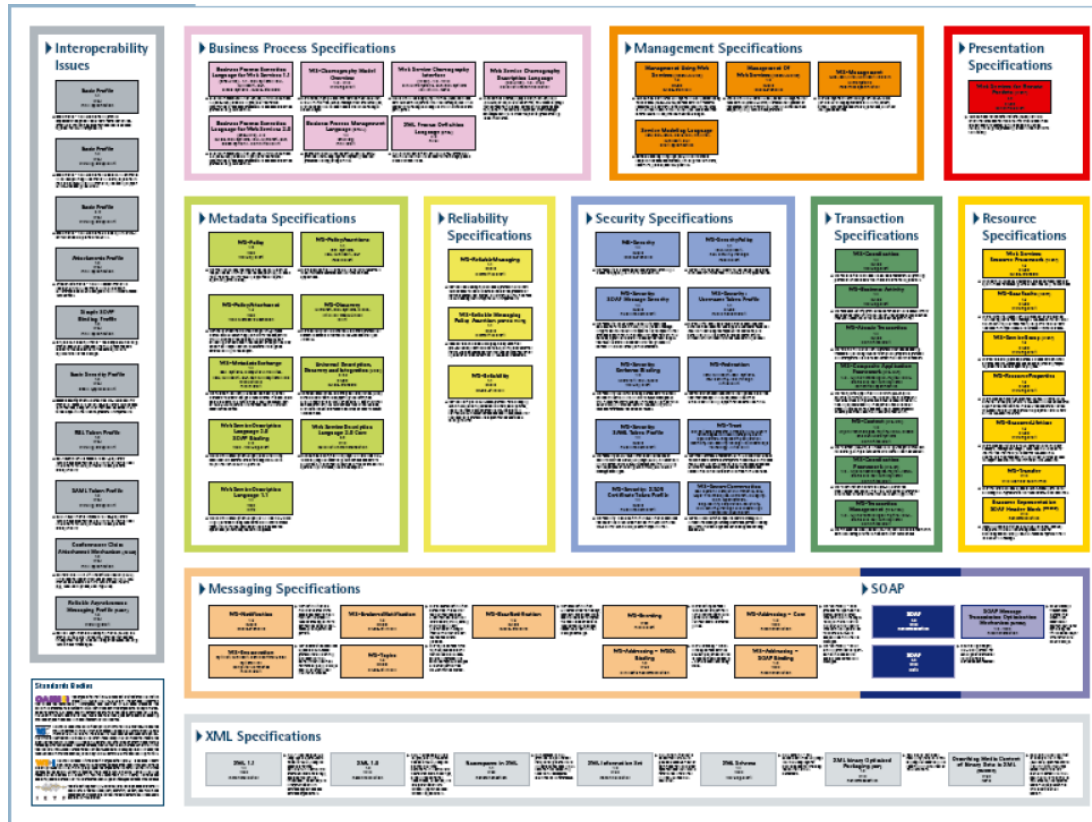


Dependencies



STANDARDS

WS-* vs. REST services



VS.



HTTP

XML

JSON





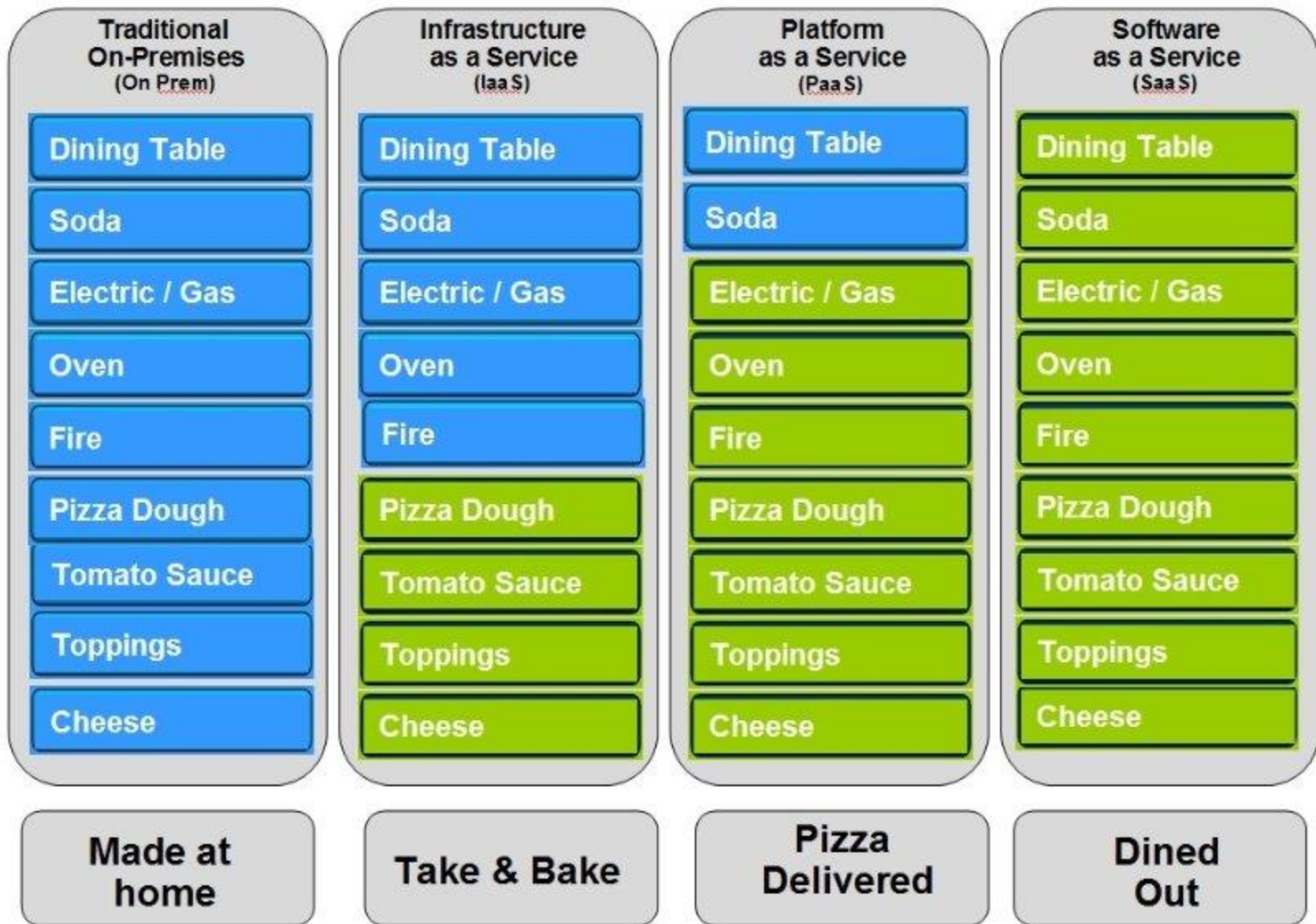
Covered syllabus (a.y. 2017/2018)

- Service-based software engineering
 - Core interoperability standards (**REST**, XML, **SOAP**, **WSDL**)
 - Software design by service composition (**WS-BPEL**, syntax & semantics)
 - Business process modelling and analysis (with workflow nets)

Service-based software engineering

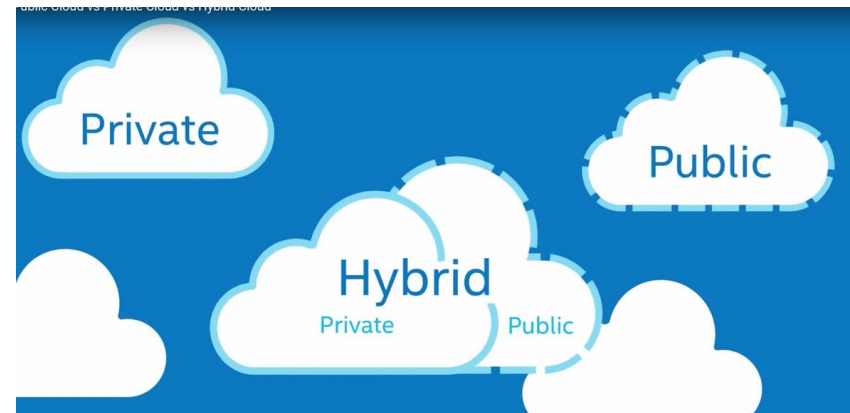
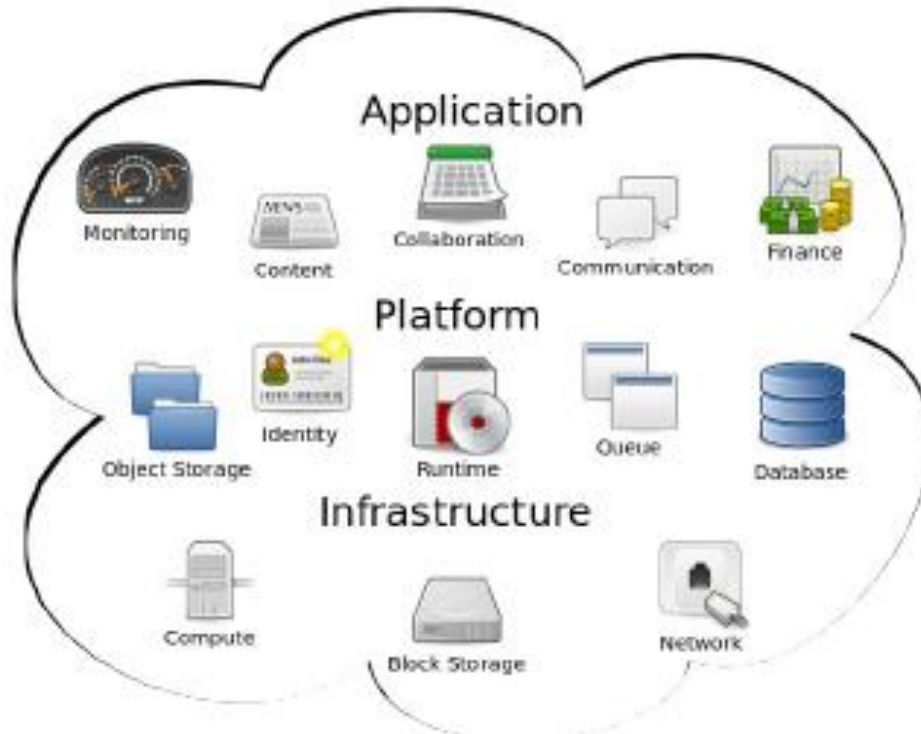
Cloud-based software engineering & DevOps

Pizza as a Service



 You Manage  Vendor Manages

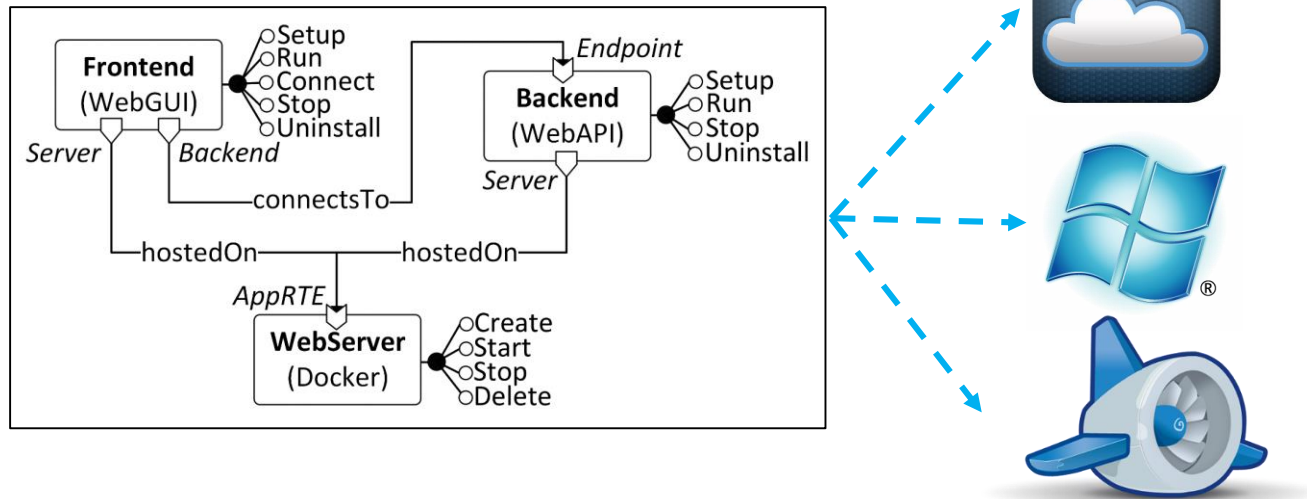
Cloud computing



Management of cloud-based apps



How to flexibly manage complex composite applications over heterogeneous cloud platforms?



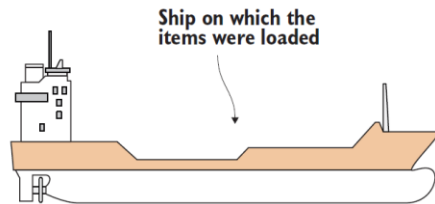
The lock-in problem

Definition - Vendor lock-in makes a customer dependent on a vendor for products or services, unable to use another vendor without substantial switching costs.

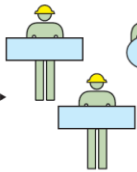


Containers

Before Containers

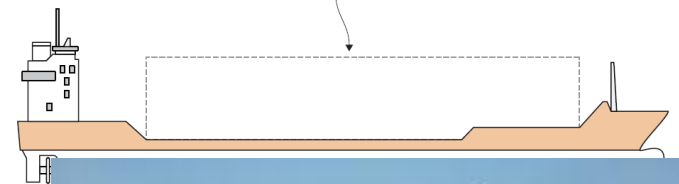


Teams of dockers required to load differently shaped items onto ship



With Containers

Ship can be designed to carry, load, and unload predictably shaped items more efficiently.

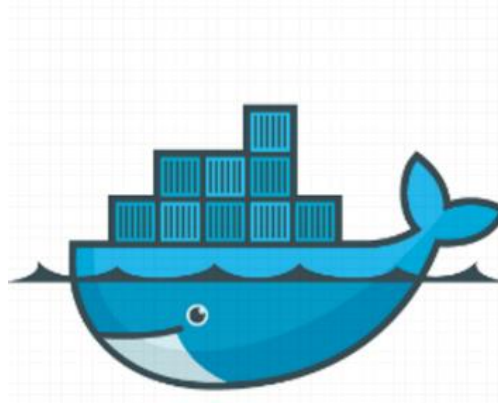


Only one operator to move

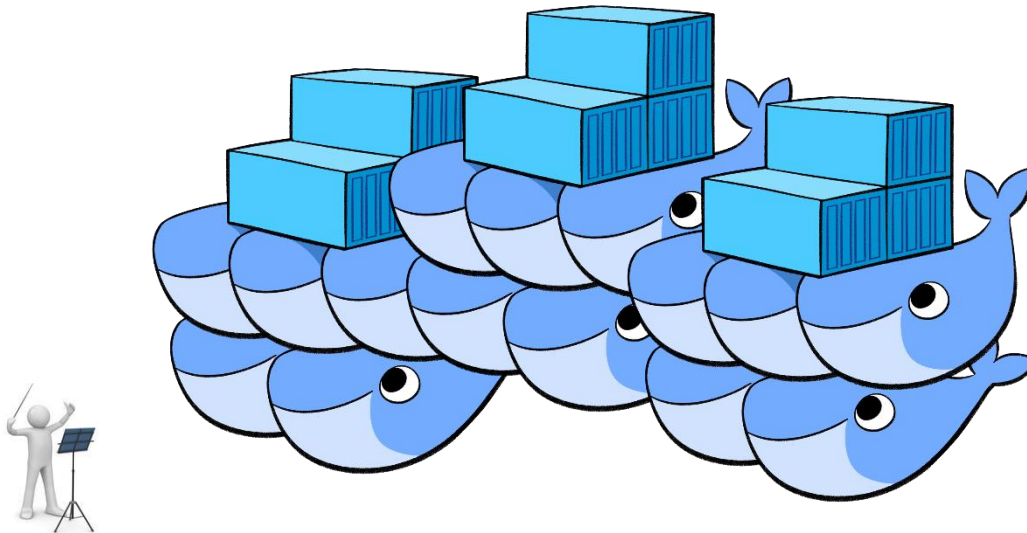


ere,

Containers



Building, shipping, and running applications, together with their dependencies, in lightweight virtual environments (containers)



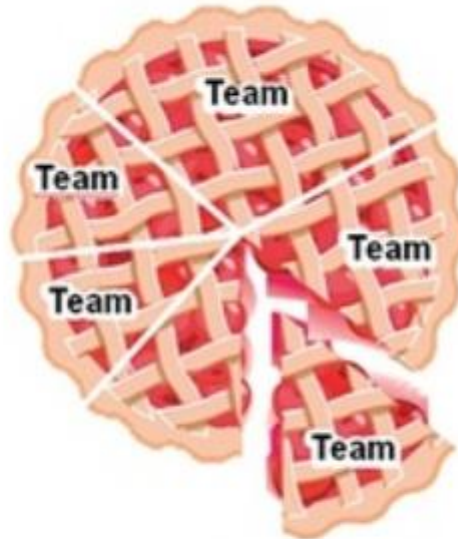
Microservices

Pre-SOA (monolithic)
Tight coupling



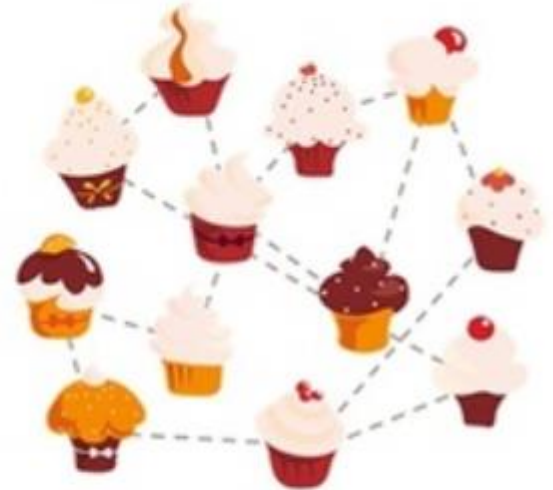
For a monolith to change, all must agree on each change. Each change has unanticipated effects requiring careful testing beforehand.

Traditional SOA
Looser coupling



Elements in SOA are developed more autonomously but must be coordinated with others to fit into the overall design.

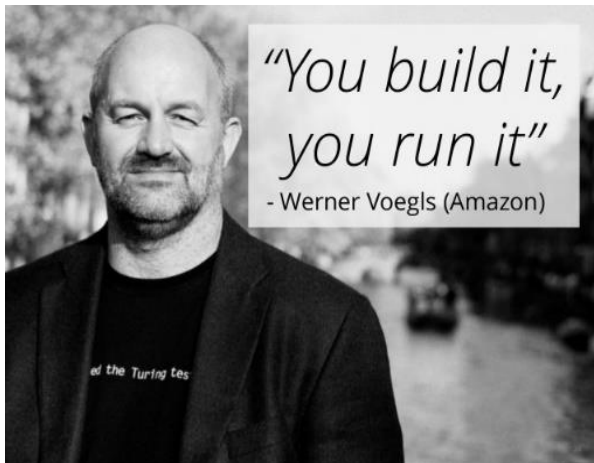
Microservices
Decoupled



Developers can create and activate new microservices without prior coordination with others. Their adherence to MSA principles makes continuous delivery of new or modified services possible.

DevOps

Projects Products

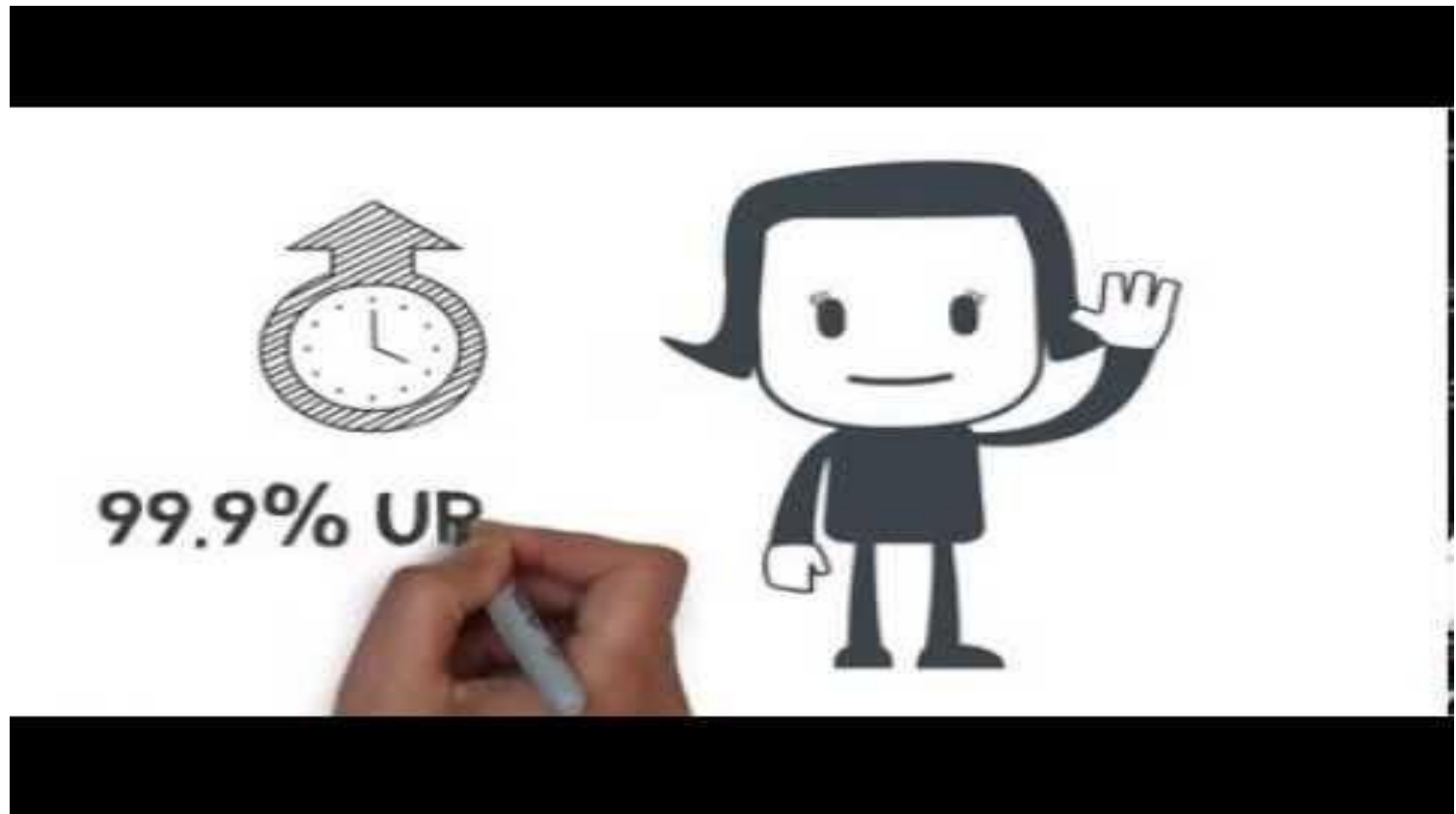


- Teams responsible for all aspects of the software they build, including operating the software 24/7
- Being woken up at 3am by your pager is certainly a powerful incentive to focus on quality when writing your code





What is DevOps?



DevOps tools

VCS (e.g., Git & GitHub)

CI&CD (e.g., Jenkins)

IaC (e.g., Puppet, Chef)

APM (e.g., NewRelic)

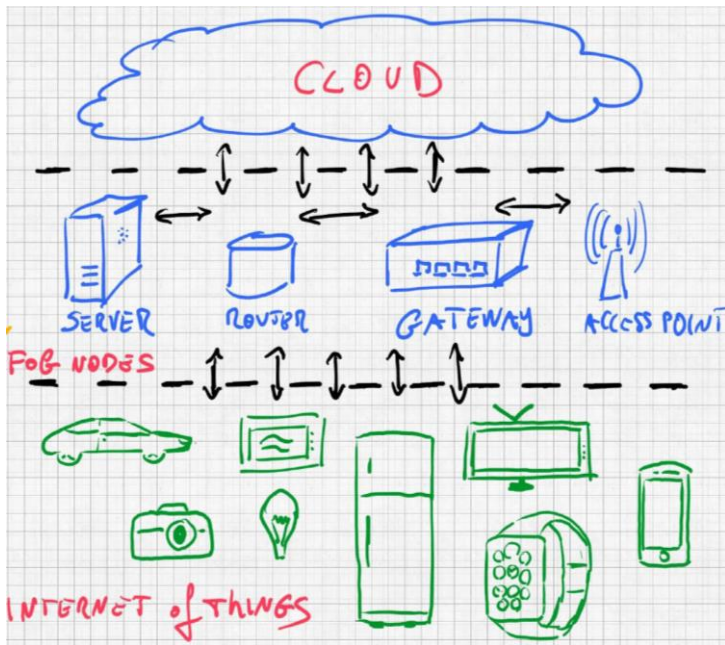
Service-based software engineering

Cloud-based software engineering & DevOps

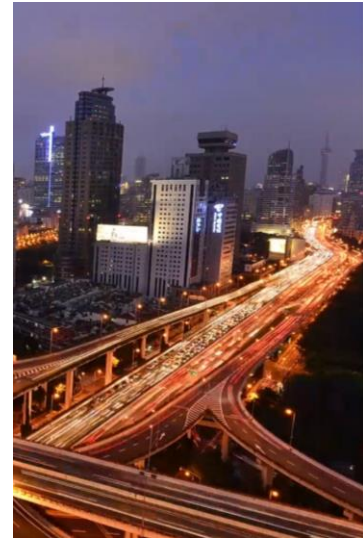
Fog computing?



- The Cloud alone cannot support the **IoT momentum**
- Need to **filter** and **process** data *before* the Cloud



Fog computing aims at extending the Cloud towards the IoT to better support **latency-sensitive** and **bandwidth-hungry** IoT applications





Drones in Fog Computing



Service-based software engineering

Cloud-based software engineering & DevOps

Fog computing?

Summary

coffee
services
QoS
SLAs
Web services
WSDL
SOAP
service composition
SOA
WS*-standards
REST

cloud
lock-in
containers
microservices
DevOps
VCS CI CD
IaC APM
fog



Service-based software engineering

Cloud-based software engineering & DevOps

Fog computing?

Summary

Course info

Instructors

- Instructor

Antonio Brogi [full professor, Ph.D.]

Department of Computer Science, University of Pisa

<http://pages.di.unipi.it/brogi/>

+ course web page

- Teaching assistant

Stefano Forti [Ph.D. student]

Department of Computer Science, University of Pisa

<http://pages.di.unipi.it/forti/>



Time

$2/3$ in class



$1/3$ in lab



BYOD?

Web

Advanced Software Engineering

Instructor: [Antonio Brogi](#) - Teaching assistant: [Stefano Forti](#)

Master's Degree in Computer Science and Master's Degree in Computer Science and Networking - a.y. 2018/2019

Current edition (a.y. 2018/2019)



[Moodle site](#)

Previous edition (a.y. 2017/2018)



[Objective](#)



[Moodle site](#)



[Classes held](#)

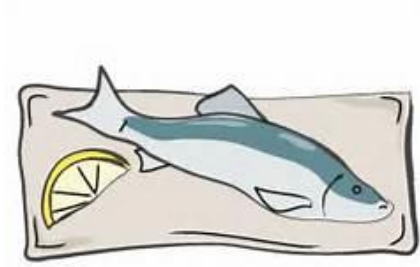


[Syllabus](#)



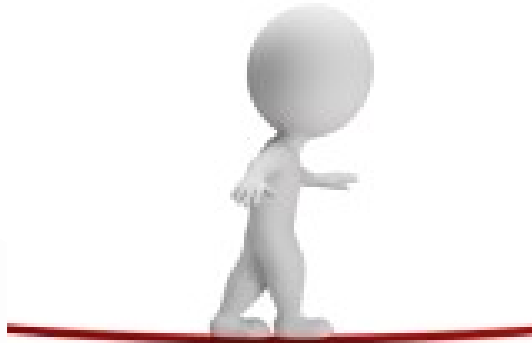
[Evaluation](#)

Challenges



Challenges

CONCEPTS



The
Big
Question!

Evaluation

(a) oral presentation on a topic chosen by student

+

(b) some questions on the syllabus

+

[opt. continuous assesement during the course]



Some stats (a.y. 2017/2018)

Average of final grades: 28,8 (!)

Students that passed the exam: 22

72% of them in first exam date

100% of them participated in continuous assessment

Students that have not yet passed the exam: 6

Failures in part (a): 4/30

Failures in part (b): 11/33 (7 failures by 3 students)

Questions?



