Basics of Cloud Computing – Lecture 8

Cloud Computing: Summary, Application Domains and Research Scope

Satish Srirama

Mobile & Cloud Lab

Outline

- Quick recap of what we have learnt as part of this course
- Research at Mobile & Cloud Lab
- Research directions for Future Generation Cloud Computing

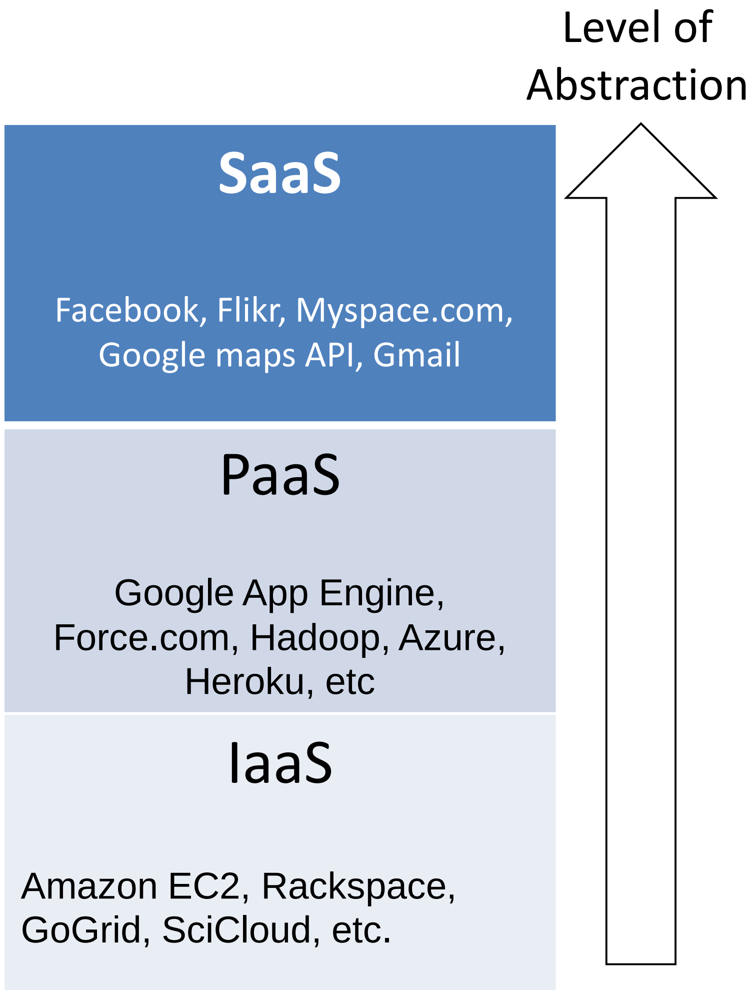
WHAT WE LEARNT IN THE COURSE!

What is Cloud Computing?

- Computing as a utility
  - Consumers pay based on their usage
- Cloud Computing characteristics
  - Illusion of infinite resources
  - No up-front cost
  - Fine-grained billing (e.g. hourly)
- Gartner: “Cloud computing is a style of computing where massively scalable IT-related capabilities are provided ‘as a service’ across the Internet to multiple external customers”

Cloud Computing - Services

- Software as a Service – SaaS
  - A way to access applications hosted on the web through your web browser
- Platform as a Service – PaaS
  - Provides a computing platform and a solution stack (e.g. LAMP) as a service
- Infrastructure as a Service – IaaS
  - Use of commodity computers, distributed across Internet, to perform parallel processing, distributed storage, indexing and mining of data
  - Virtualization

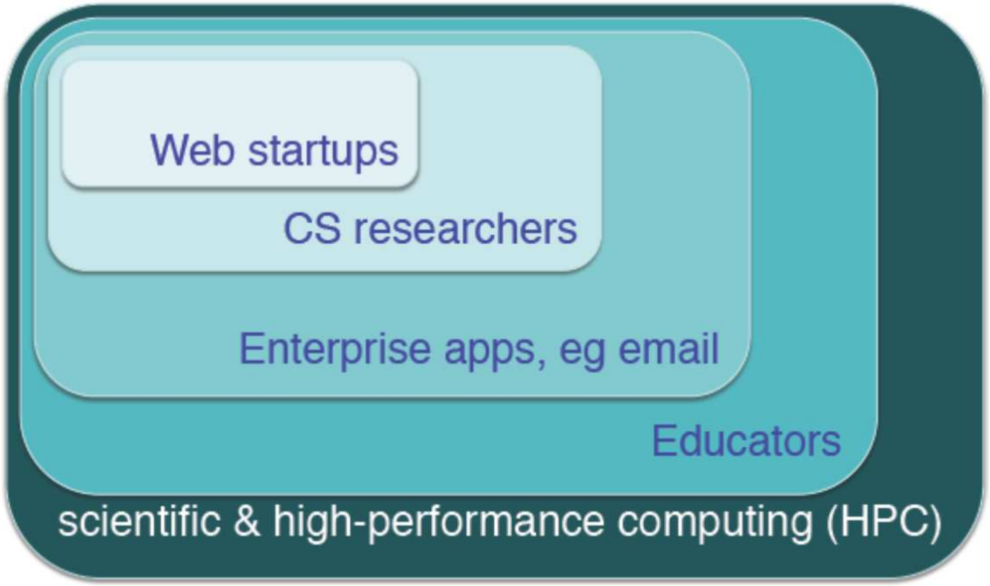


Cloud Computing - Themes

- Massively scalable
- On-demand & dynamic
- Only use what you need - Elastic
  - No upfront commitments, use on short term basis
- Accessible via Internet, location independent
- Transparent
  - Complexity concealed from users, virtualized, abstracted
- Service oriented
  - Easy to use Service Level Agreements

Cloud Computing Progress

- Short term and long term implications of cloud
- Economics of cloud users and cloud providers
- Challenges and opportunities offered by cloud computing



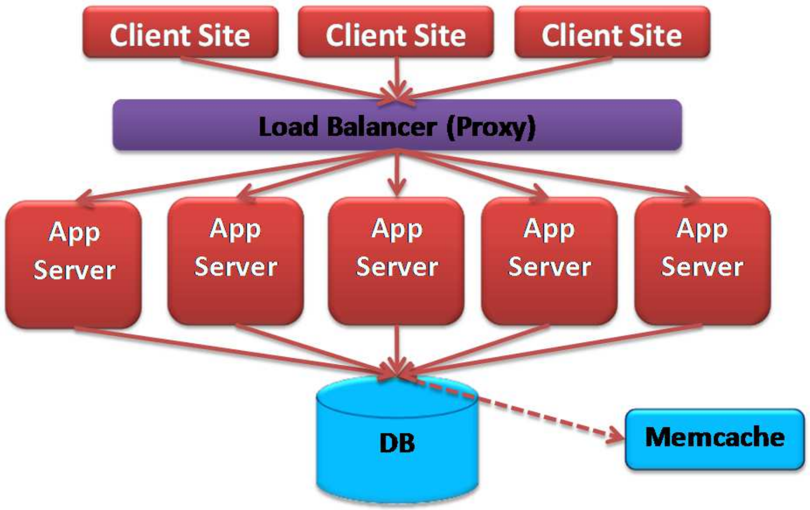
[Armando Fox, 2010]

Cloud Providers

- Amazon Web Services
  - EC2, S3, EBS, Elastic Load Balancing, Amazon Auto Scale, Amazon CloudWatch, IAM, CloudFormation, Data Pipelines, Data migration services etc.
- Private Cloud enabling technologies
  - Eucalyptus
  - OpenStack
    - Worked with SciCloud

Scaling Applications on the Cloud

- Two basic models of scaling
  - Vertical scaling, aka Scale-up
  - Horizontal scaling, aka Scale-out
- Scaling Enterprise Applications in the Cloud
- Load balancing
  - Types and algorithms
- Autoscaling



Economics of Cloud Providers

- Cloud Computing providers bring a shift from high reliability/availability servers to commodity servers
  - At least one failure per day in large datacenter
- Why?
  - Significant economic incentives
    - much lower per-server cost
- Caveat: User software has to adapt to failures
  - Very hard problem!
- Solution: Replicate data and computation
  - MapReduce & Distributed File System

MapReduce

- Programmers specify two functions:
  map (k, v) → <k’, v’>*
  reduce (k’, v’) → <k’, v’>*
  – All values with the same key are reduced together
- The execution framework handles everything else…
- Not quite…usually, programmers also specify:
  partition (k’, number of partitions) → partition for k’
  – Often a simple hash of the key, e.g., hash(k’) mod n
  – Divides up key space for parallel reduce operations
  combine (k’, v’) → <k’, v’>*
  – Mini-reducers that run in memory after the map phase
  – Used as an optimization to reduce network traffic

Hadoop Processing Model

- Create or allocate a cluster
- Put data onto the file system (HDFS)
  - Data is split into blocks
  - Replicated and stored in the cluster
- Run your job
  - Copy Map code to the allocated nodes
    - Move computation to data, not data to computation
  - Gather output of Map, sort and partition on key
  - Run Reduce tasks
- Results are stored in the HDFS

MapReduce Examples

- Distributed Grep
- Count of URL Access Frequency
- Reverse Web-Link Graph
- Inverted Index
- Distributed Sort

Synchronization in Hadoop

- Approach 1: turn synchronization into an ordering problem
  - Sort keys into correct order of computation
  - Partition key space so that each reducer gets the appropriate set of partial results
  - Hold state in reducer across multiple key-value pairs to perform computation
  - Illustrated by the “pairs” approach in calculating conditional probability of words
- Approach 2: construct data structures that “bring the pieces together”
  - Each reducer receives all the data it needs to complete the computation
  - Illustrated by the “stripes” approach

Platform as a Service -PaaS

- Complete platform for hosting applications in Cloud
- The underlying infrastructure & software environment is managed for you
- Google App Engine
- Advantages:
  - User does not have to manage low level computing resources and services
  - Provider handles most of the non functional requirements of your applications
- Disadvantages:
  - Not in full control over computing resources, software and library versions, service configuration etc.
  - Vendor lock-in

Serverless computing

- Newer workloads are a better fit for event driven programming
  - Execute application logic in response to database triggers
  - Execute app logic in response to sensor data
  - Execute app logic in response to scheduled tasks etc.
- Serverless in a nutshell
  - Event-action platforms to execute code in response to events
- Applications are charged by compute time (millisecond) rather than by reserved resources
- Greater linkage between cloud resources used and business operations executed

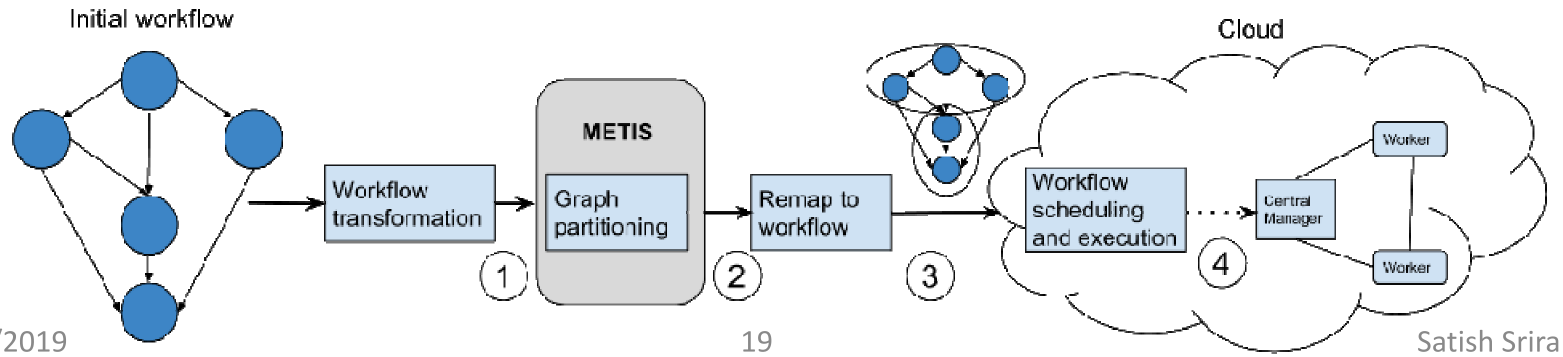
CLOUD COMPUTING: APPLICATIONS & RESEARCH SCOPE

Scope of the Cloud Applications

- Already discussed scaling enterprise applications on the cloud
- Scientific Computing on the Cloud
  - Public clouds provide very convenient access to computing resources
  - On-demand and in real-time
  - As long as you can afford them
  - Research at scale and Cost-to-value of experiments
- High performance computing (HPC) on cloud
  - Virtualization and communication latencies are major hindrances [Srirama et al, SPJ 2011; Batrashev et al, HPCS 2011]
    - Things have improved significantly over the years
    - Containers improved the scenario further

Migrating Scientific Workflows to the Cloud [Srirama and Viil, HPCC 2014]

- Workflow can be represented as weighted directed acyclic graph (DAG)
- Partitioning the workflow into groups with graph partitioning techniques
  - Such that the sum of the weights of the edges connecting to vertices in different groups is minimized
  - Utilized Metis' multilevel k-way partitioning



Migrating Scientific Workflows to the Cloud - continued

- Scheduling the workflows with tools like Pegasus
  - Considered peer-to-peer file manager (Mule) for Pegasus
- Framework for Automated Partitioning and Execution of Scientific Workflows in the Cloud [Viil and Srirama, JSC 2018]
  - Includes auto-scaling and dynamic deployment with CloudML

Adapting Computing Problems to MapReduce

- Designed a classification on how the algorithms can be adapted to MapReduce [Srirama et al, FGCS 2012]
  - Algorithm → single MapReduce job
    - Monte Carlo, RSA breaking
  - Algorithm → *n* MapReduce jobs
    - CLARA (Clustering), Matrix Multiplication
  - Each iteration in algorithm → single MapReduce job
    - PAM (Clustering)
  - Each iteration in algorithm → *n* MapReduce jobs
    - Conjugate Gradient
- Applicable especially for Hadoop MapReduce

MapReduce Limitations

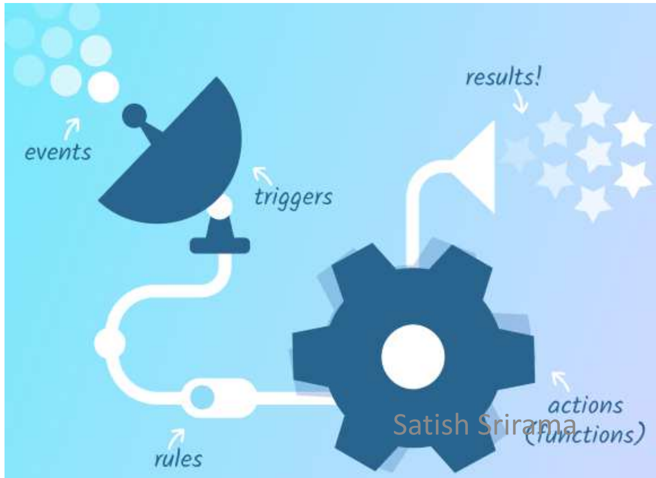
- MapReduce has serious issues with iterative algorithms
  - Long „*start up*“ and „*clean up*“ times **~17** seconds
  - No way to keep important data in memory between MapReduce job executions
  - At each iteration, all data is read again from HDFS and written back there at the end
  - Results in a significant overhead in every iteration

Alternative Approaches

- Restructuring algorithms into non-iterative versions
  - CLARA instead of PAM [Jakovits & Srirama, Nordicloud 2013]
- Alternative MapReduce implementations that are designed to handle iterative algorithms [Jakovits and Srirama, HPCS 2014]
  - E.g. Twister, HaLoop, Spark
- Alternative distributed computing models
  - Bulk Synchronous Parallel model [Valiant, 1990] [Jakovits et al, HPCS 2013]

EU H2020 -RADON

- Rational decomposition and orchestration for serverless computing
  - Jan 2019 – Jun 2021
- Goal
  - Creating a DevOps framework to create and manage microservices-based applications
  - Tools that facilitate in designing and orchestrating data pipeline applications that involve serverless entities
  - OASIS - Topology and Orchestration Specification for Cloud Applications specification (TOSCA)
- Case studies
  - IoT application from healthcare
  - Tourism



Mobile Applications

- One can do interesting things on mobiles directly
  - Today's mobiles are far more capable
  - Location-based services (LBSs), mobile social networking, mobile commerce, context-aware services etc.
- It is also possible to make the mobile a service provider
  - Mobile web service provisioning [Srirama et al, ICIW 2006; Srirama and Paniagua, MS 2013]
  - Challenges in security, scalability, discovery and middleware are studied [Srirama, PhD 2008]
  - Mobile Social Network in Proximity [Chang et al, PMC 2014]

However, we still have not achieved

- Longer battery life
  - Battery lasts only for 1-2 hours for continuous computing
- Same quality of experience as on desktops
  - Weaker CPU and memory
  - Storage capacity
- Still it is a good idea to take the support of external resources for building resource intensive mobile applications

Mobile Cloud

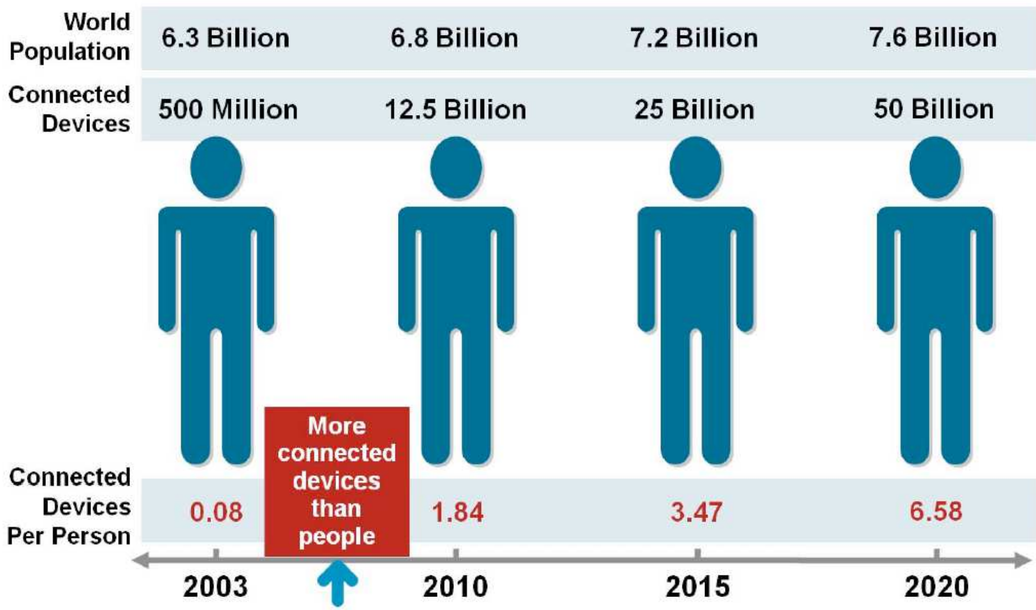
- Harness cloud computing resources from mobile devices

- Binding models
  - Task delegation [Flores and Srirama, JSS 2014]
  - Mobile code offloading [Flores et al, IEEE Communications Mag 2015; Zhou et al, TSC 2017]

- Ideal Mobile Cloud based system should take advantage of some of the key intrinsic characteristics of cloud efficiently
  - Elasticity & AutoScaling
  - Utility computing models
  - Parallelization (e.g., using MapReduce)

Internet of Things (IoT)

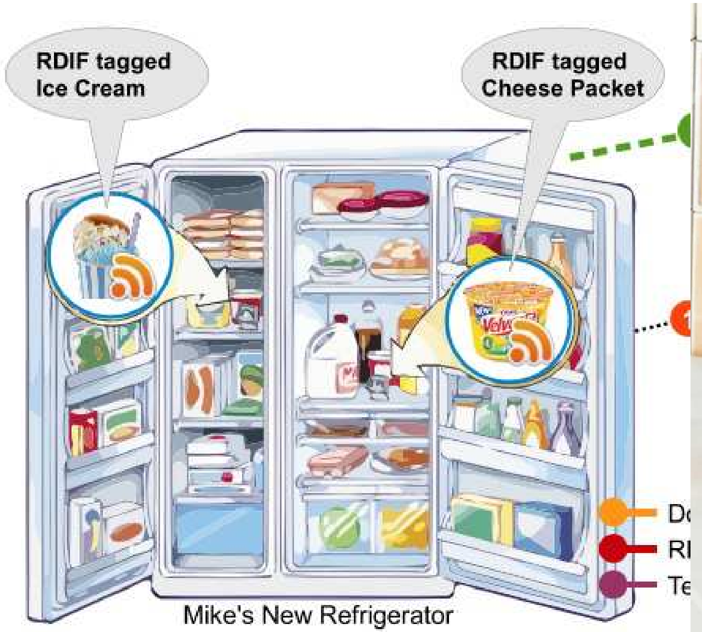
- *"The Internet of Things allows people and things to be connected* ***Anytime***, ***Anyplace***, *with* ***Anything*** *and* ***Anyone***, *ideally using* ***Any path/network*** *and* ***Any service***." [European Research Cluster on IoT
- More connected (
- Cisco believes the trillion by 2025

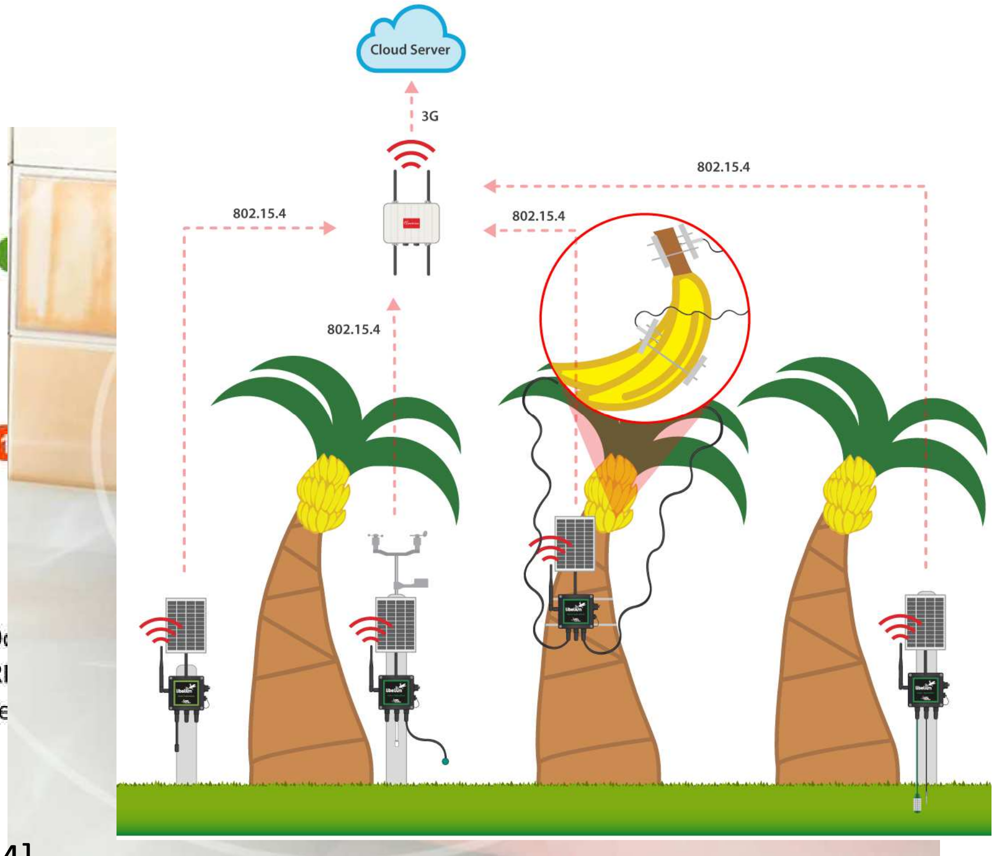


Source: Cisco IBSG, April 2011

IoT - Scenarios

- Environment Protection
- Smart Home
-
-

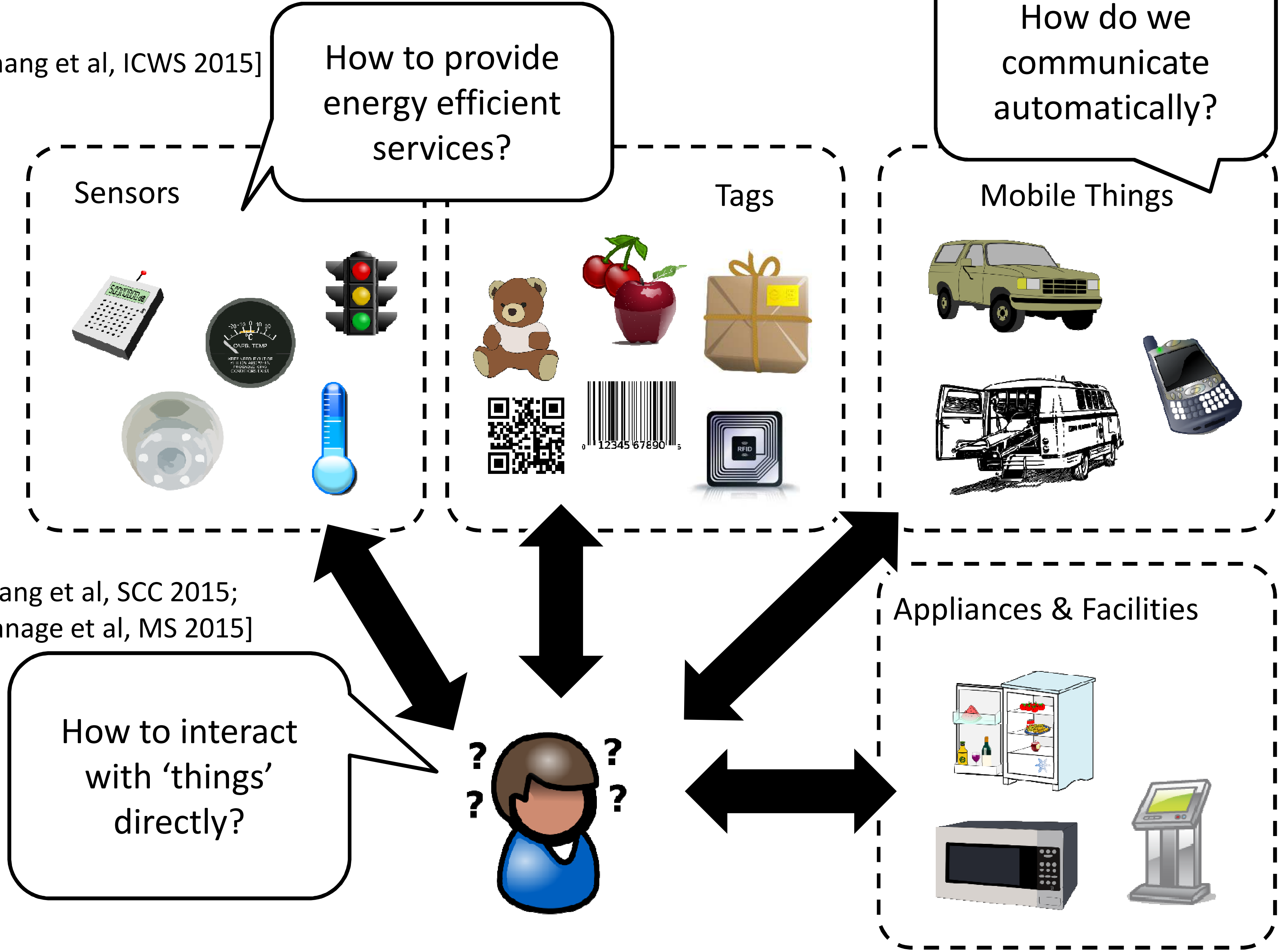




[Kip Compton]
[Perera et al, TETT 2014]
[http://www.libelium.com/improving-banana-crops-production-and-agricultural-sustainability-in-colombia-using-sensor-networks/]

Internet of Things – Challenges

[Chang et al, ICWS 2015]



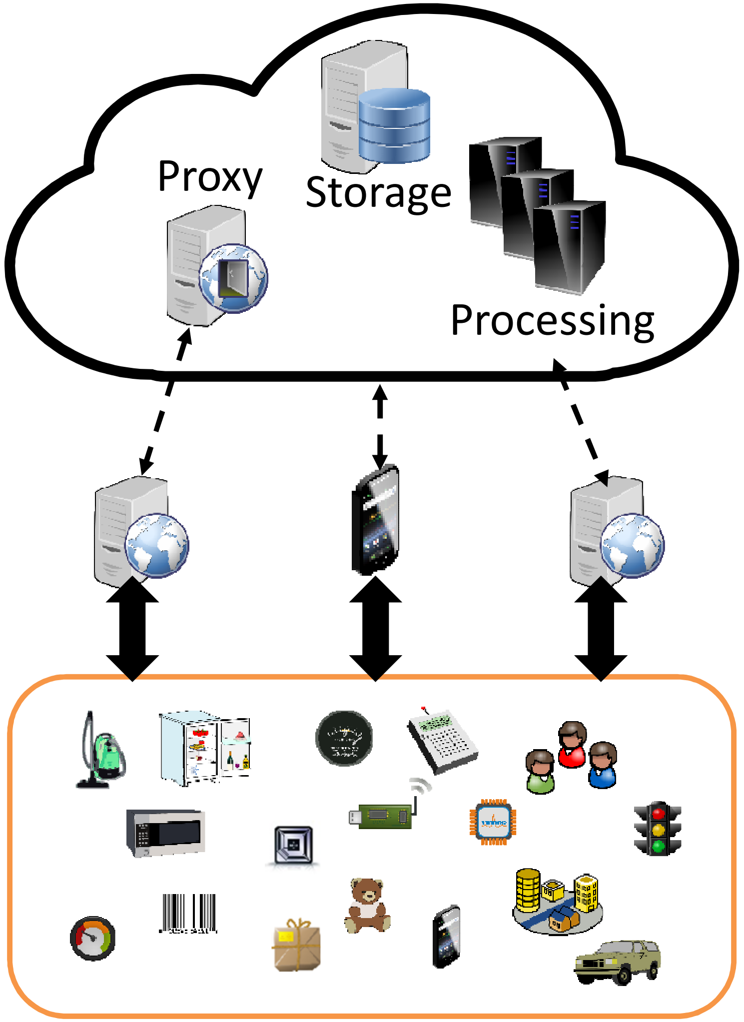
[Chang et al, SCC 2015; Liyanage et al, MS 2015]

Cloud-centric IoT

Remote Cloud-based processing

Connectivity nodes & Embedded processing

Sensing and smart devices



[Srirama, CSIICT 2017]

IoT Data Processing on Cloud

- Enormous amounts of unstructured data
  - In Zetabytes (10^{21} bytes) by 2020 [TelecomEngine]
  - Has to be properly stored, analysed and interpreted and presented
- Big data acquisition and analytics
  - Is MapReduce sufficient?
    - MapReduce is not good for iterative algorithms [Srirama et al, FGCS 2012]
- In addition to big data, IoT mostly deals with big streaming data
  - Message queues such as Apache Kafka to buffer and feed the data into stream processing systems such as Apache Storm
  - Apache Spark streaming

[Distributed Data Processing on the Cloud - LTAT.06.005 (Fall 2018)]

Issues with Cloud-centric IoT

- Latency issues for applications with sub-second response requirements
- Certain scenarios do not let the data move to cloud
- Fog computing [Chang et al, AINA 2017]
  - Processing across all the layers, including network switches/routers
- Challenges in Fog computing
  - Mobility, task migration, discovery, scalability and containerization [Soo et al, IJMCMC 2017; Chang et al, IEEE Computer 2017]
  - QoE-aware application placement across Fog topology [Mahmud et al, JPDC 2018]

Research Roadmap – IoT & Fog Computing

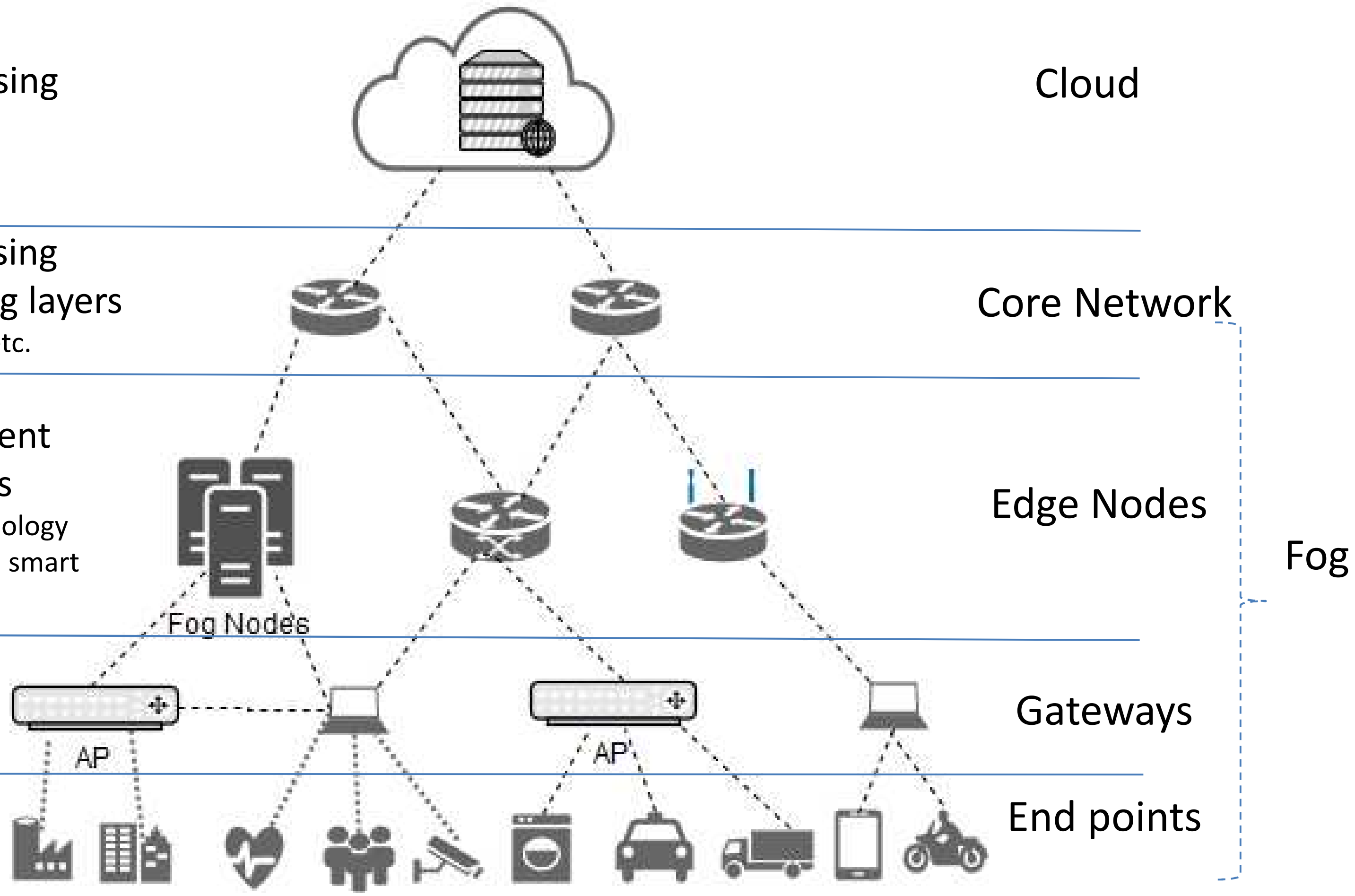
Distributed data processing on the Cloud
E.g. MapReduce, Spark

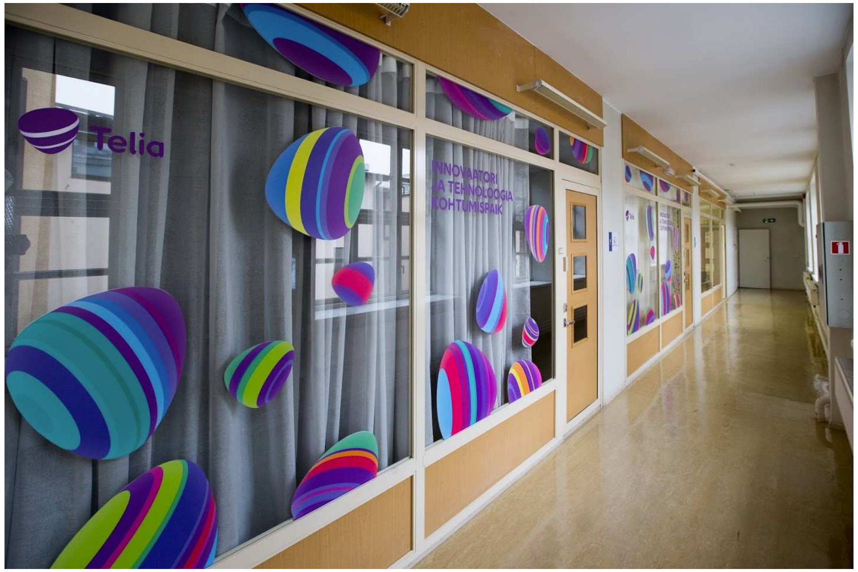
Distributed data processing across the Cloud and Fog layers
E.g. Personalized data, privacy etc.

Fog topology management and scheduling the tasks
E.g. tasks run across the fog topology such as stream data processing, smart streetlights etc.

Edge analytics
E.g. filter, error detection, consolidation etc.

Intelligent sensors
E.g. vehicular networks





Mobile & Cloud Lab

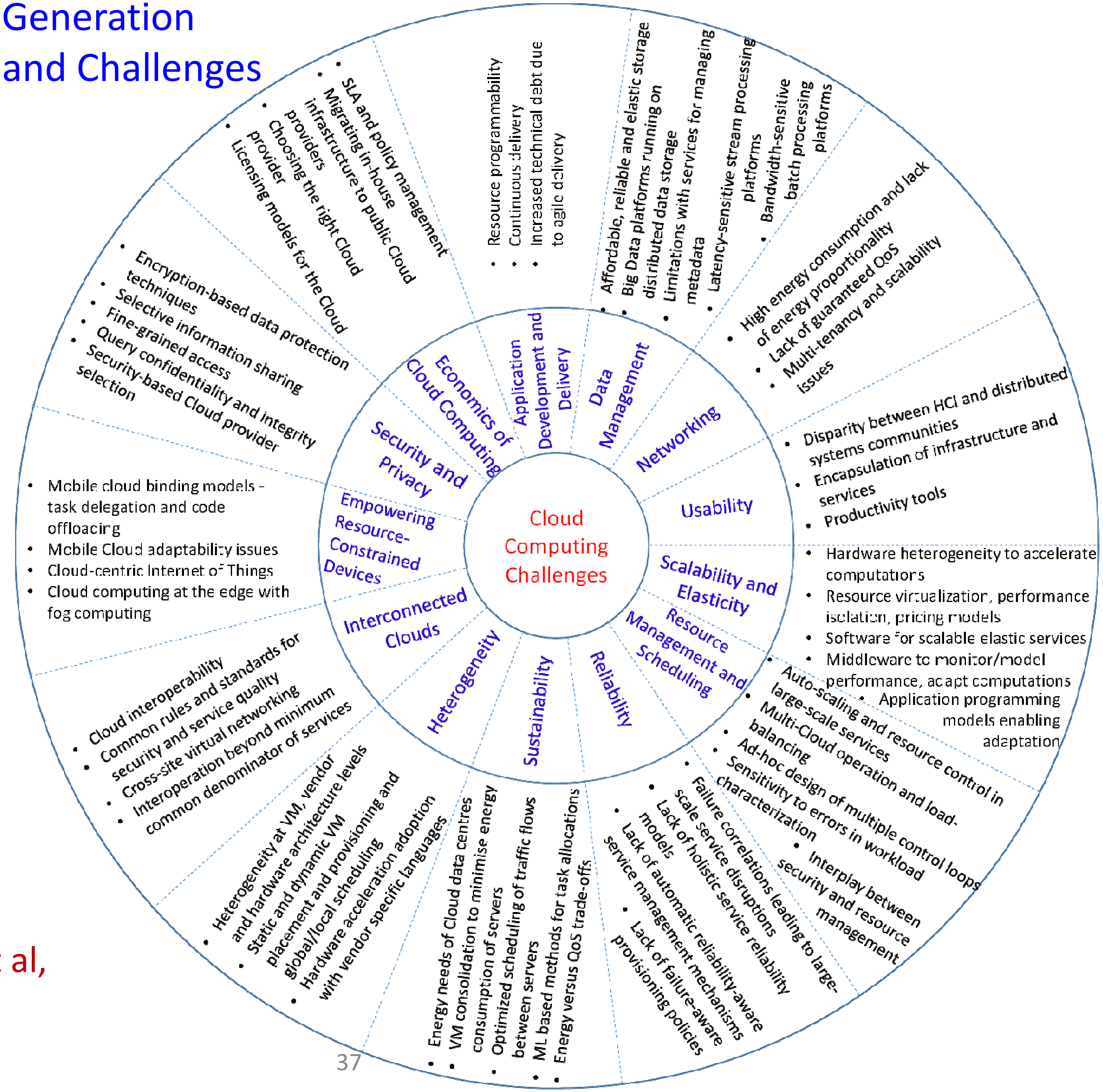
Telia

WE ALWAYS WELCOME NEW IDEAS!

Research @ Mobile & Cloud Lab

- Scientific computing on the cloud
- Classification on how the algorithms can be adapted to MR
- Limitations of Hadoop MapReduce
  - Alternatives
- Mobile Cloud
- Cloud-centric Internet of Things
- Fog computing

A Manifesto for Future Generation Cloud Computing: SOA and Challenges



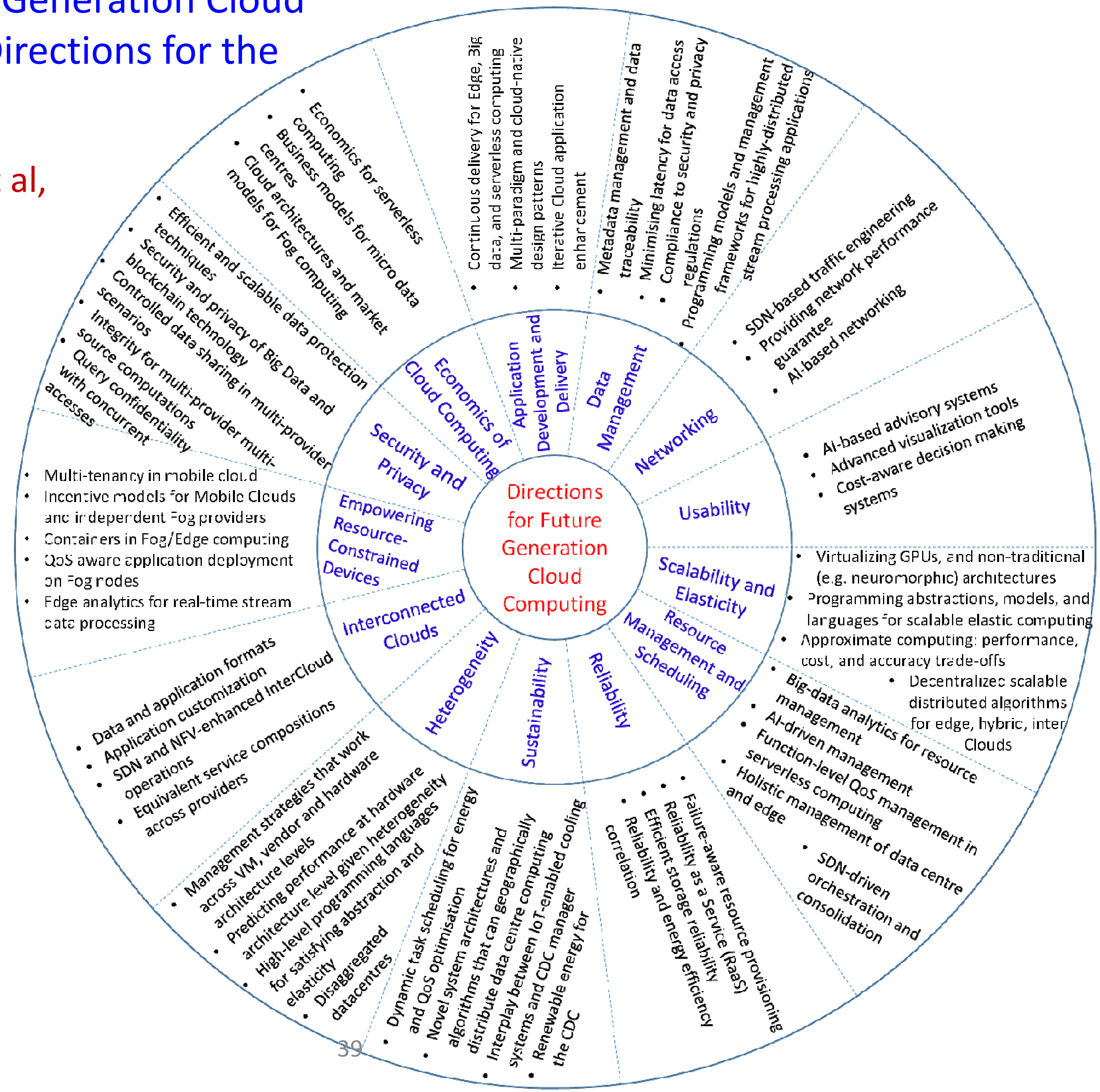
[Buyya, Srirama, Casale et al, ACM CSUR 2019]

Emerging trends and impact areas for cloud

- Containers
- Fog Computing
- Big Data
- Serverless Computing
- Software-defined Cloud Computing
- Blockchain
- Machine and Deep Learning

A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade

[Buyya, Srirama, Casale et al, ACM CSUR 2019]



This week in lab

- Work with cloud functions
  - Cloud Functions in IBM Bluemix (Managed OpenWhisk service)

Examination

- Rooms and exact times will be announced soon
- Exam I : Tuesday, 16.04.2019, 12:00 – 15:00 J. Liivi 2 - 511
- Exam II : Wednesday, 17.04.2019, 12:00 – 15:00 in room Ulikooli 17- 218

Preparation for Examination

- One of the earlier exam papers is kept online
- Slides are mostly self sufficient
- References are mentioned for further reading
- Mainly focus at keywords

References

- Basics of Cloud Computing – Lectures https://courses.cs.ut.ee/2016/cloud/spring/Main/Lectures
- [Flores et al, IEEE Communications Mag 2015] H. Flores, P. Hui, S. Tarkoma, Y. Li, S. N. Srirama, R. Buyya: Mobile Code Offloading: From Concept to Practice and Beyond, IEEE Communications Magazine, ISSN: 0163-6804, 53(3):80-88, 2015. IEEE. DOI:10.1109/MCOM.2015.7060486
- [Chang et al, SCC 2015]C. Chang, S. N. Srirama, J. Mass: A Middleware for Discovering Proximity-based Service-Oriented Industrial Internet of Things, 12th IEEE International Conference on Services Computing (SCC 2015), June 27 - July 2, 2015, pp. 130-137. IEEE.
- [Liyanage et al, MS 2015] M. Liyanage, C. Chang, S. N. Srirama: Lightweight Mobile Web Service Provisioning for Sensor Mediation, 4th International Conference on Mobile Services (MS 2015), June 27 - July 2, 2015, pp. 57-64. IEEE
- [Chang et al, ICWS 2015] C. Chang, S. Loke, H. Dong, F. Salim, S. N. Srirama, M. Liyanage, S. Ling: An Energy-Efficient Inter-organizational Wireless Sensor Data Collection Framework, The IEEE 22nd International Conference on Web Services (ICWS 2015), June 27 - July 2, 2015, pp. 639-646. IEEE.
- [Flores and Srirama, JSS 2014] H. Flores, S. N. Srirama: Mobile Cloud Middleware, Journal of Systems and Software, ISSN: 0164-1212, 92(1):82-94, 2014. Elsevier. DOI: 10.1016/j.jss.2013.09.012.
- [Chang et al, PMC 2014] C. Chang, S. N. Srirama, S. Ling: Towards an Adaptive Mediation Framework for Mobile Social Network in Proximity, Pervasive and Mobile Computing Journal, MUCS Fast track, ISSN: 1574-1192, 12:179-196, 2014. Elsevier. DOI: 10.1016/j.pmcj.2013.02.004.
- [Srirama and Paniagua, MS 2013] S. N. Srirama, C. Paniagua: Mobile Web Service Provisioning and Discovery in Android Days, The 2013 IEEE International Conference on Mobile Services (MS 2013), June 27 - July 02, 2013, pp. 15-22. IEEE.
- [Flores and Srirama, MCS 2013] H. Flores, S. N. Srirama: Adaptive Code Offloading for Mobile Cloud Applications: Exploiting Fuzzy Sets and Evidence-based Learning, The Fourth ACM Workshop on Mobile Cloud Computing and Services (MCS 2013) @ MobiSys 2013, June 25-28, 2013, pp. 9-16. ACM.
- [Srirama et al, FGCS 2012] S. N. Srirama, P. Jakovits, E. Vainikko: Adapting Scientific Computing Problems to Clouds using MapReduce, Future Generation Computer Systems Journal, 28(1):184-192, 2012. Elsevier press. DOI 10.1016/j.future.2011.05.025.
- [Srirama, PhD 2008] S. N. Srirama: Mobile Hosts in Enterprise Service Integration, PhD thesis, RWTH Aachen University, September, 2008.
- [Srirama et al, ICIW 2006] S. N. Srirama, M. Jarke, W. Prinz: Mobile Web Service Provisioning, Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT-ICIW 2006), February 23-25, 2006, pp. 120-125. IEEE Computer Society Press.
- [Srirama, CSIICT 2017] S. N. Srirama: Mobile Web and Cloud Services Enabling Internet of Things, Special Issue ICAC 2016 of CSIT, CSI Transactions on ICT, ISSN: 2277-9078, 5(1):109-117, 2017. Springer. DOI: 10.1007/s40012-016-0139-3
- [Zhou et al, TSC 2017] B. Zhou, A. V. Dastjerdi, R. N. Calheiros, S. N. Srirama, R. Buyya: mCloud: A Context-aware Offloading Framework for Heterogeneous Mobile Cloud, IEEE Transactions on Services Computing, ISSN: 1939-1374, 10(5):797-810, 2017. IEEE. DOI: 10.1109/TSC.2015.2511002
- [Mahmud et al, JPDC 2018] R. Mahmud, S. N. Srirama, R. Kotagiri, R. Buyya: Quality of Experience (QoE)-aware Placement of Applications in Fog Computing Environments, Journal of Parallel and Distributed Computing, ISSN: 0743-7315, 2018. Elsevier. (In Print)
- [Soo et al, IJMCMC 2017] S. Soo, C. Chang, S. Loke, S. N. Srirama: Proactive Mobile Fog Computing using Work Stealing: Data Processing at the Edge, International Journal of Mobile Computing and Multimedia Communications (IJMCMC), ISSN: 1937-9412, 8(4):1-19, 2017. IGI Global.
- [Chang et al, IEEE Computer] C. Chang, S. N. Srirama, R. Buyya: Indie Fog: An Efficient Fog-Computing Infrastructure for the Internet of Things, IEEE Computer, ISSN: 0018-9162, 50(9):92-98, 2017. IEEE. DOI: 10.1109/MC.2017.3571049
- [Liyanage et al, PDCAT 2016] M. Liyanage, C. Chang, S. N. Srirama: mePaaS: Mobile-Embedded Platform as a Service for Distributing Fog Computing to Edge Nodes, The 17th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT-16), Guangzhou, China, December 16-18, 2016, pp. 73-80. IEEE.
- [Perera et al, TETT 2014] Perera, C., Zaslavsky, A., Christen, P., & Georgakopoulos, D. (2014). Sensing as a service model for smart cities supported by internet of things. Transactions on Emerging Telecommunications Technologies, 25(1), 81-93.
- [Chang et al, AINA 2017] C. Chang, M. Liyanage, S. Soo, S. N. Srirama: Fog Computing as a Resource-Aware Enhancement for Vicinal Mobile Mesh Social Networking, The 31-st IEEE International Conference on Advanced Information Networking and Applications (AINA-2017), Taipei, Taiwan, March 27-29, 2017, pp. 894-901. IEEE.
- [Buyya and Srirama et al, CSUR 2019] R. Buyya, S.N. Srirama, G. Casale, et al: A Manifesto for Future Generation Cloud Computing: Research Directions for the Next Decade, ACM Computing Surveys, ISSN 0360-0300, 51(5):105, 38 pages, 2019. ACM Press, New York, USA. DOI: https://doi.org/10.1145/3241737
- [Vill and Srirama, JSC 2018] J. Viil, S. N. Srirama: Framework for Automated Partitioning and Execution of Scientific Workflows in the Cloud, The Journal of Supercomputing, ISSN: 0920-8542, 74(6):2656–2683, 2018. Springer. DOI: 10.1007/s11227-018-2296-7