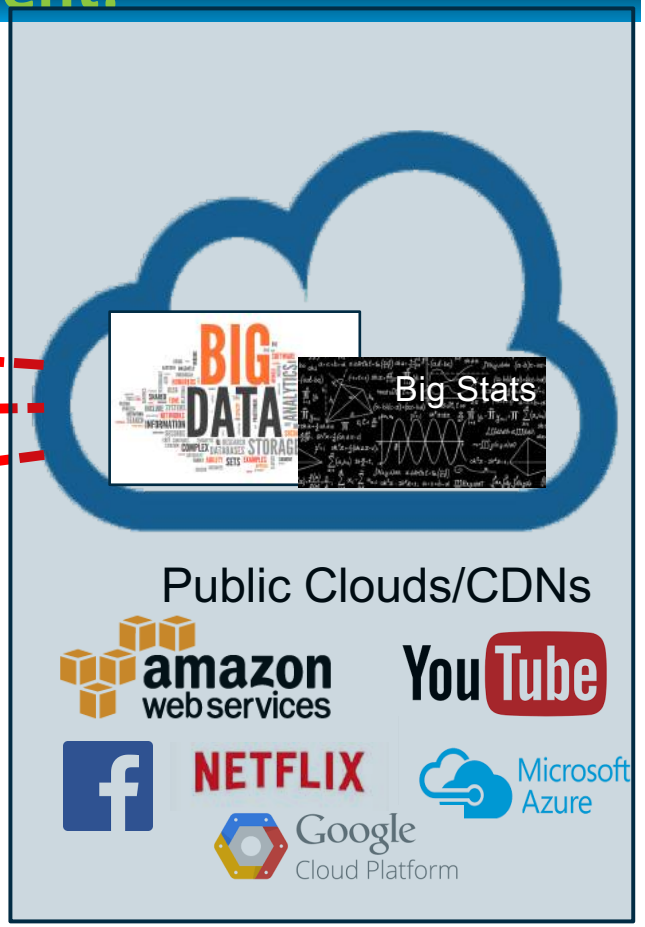
CSPOT: Portable Multi-scale Functions-as-a-Service for IoT

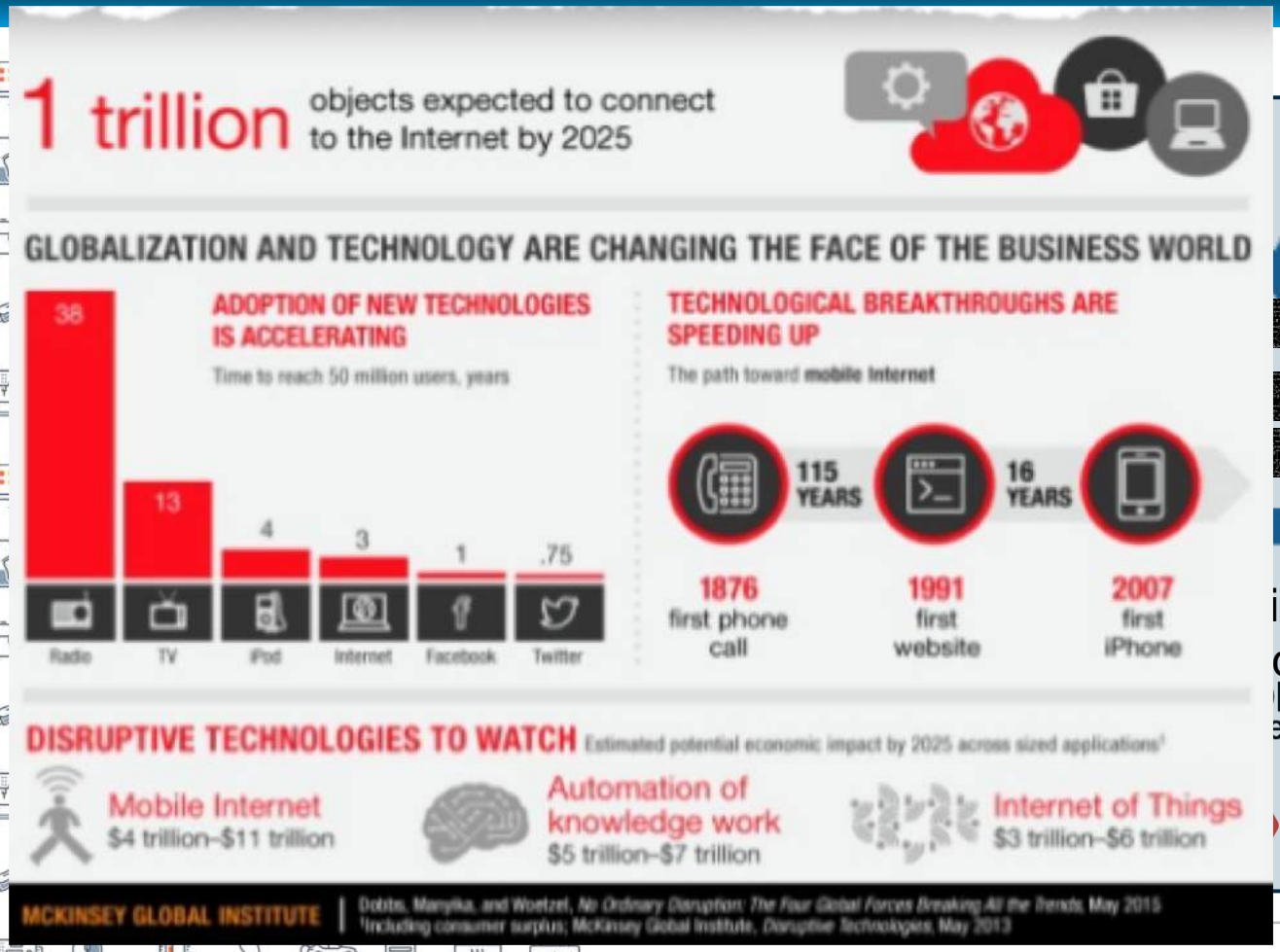
Rich Wolski, Chandra Krintz, Fatih Bakir, Gareth George, Wei-tsung Lin
Computer Science Department
University of California, Santa Barbara
November 8, 2019

Cloud Today = E-Commerce & Entertainment:
Data Moves From Cloud to User Devices



- Download (read) dominated: streaming entertainment/content

IoT

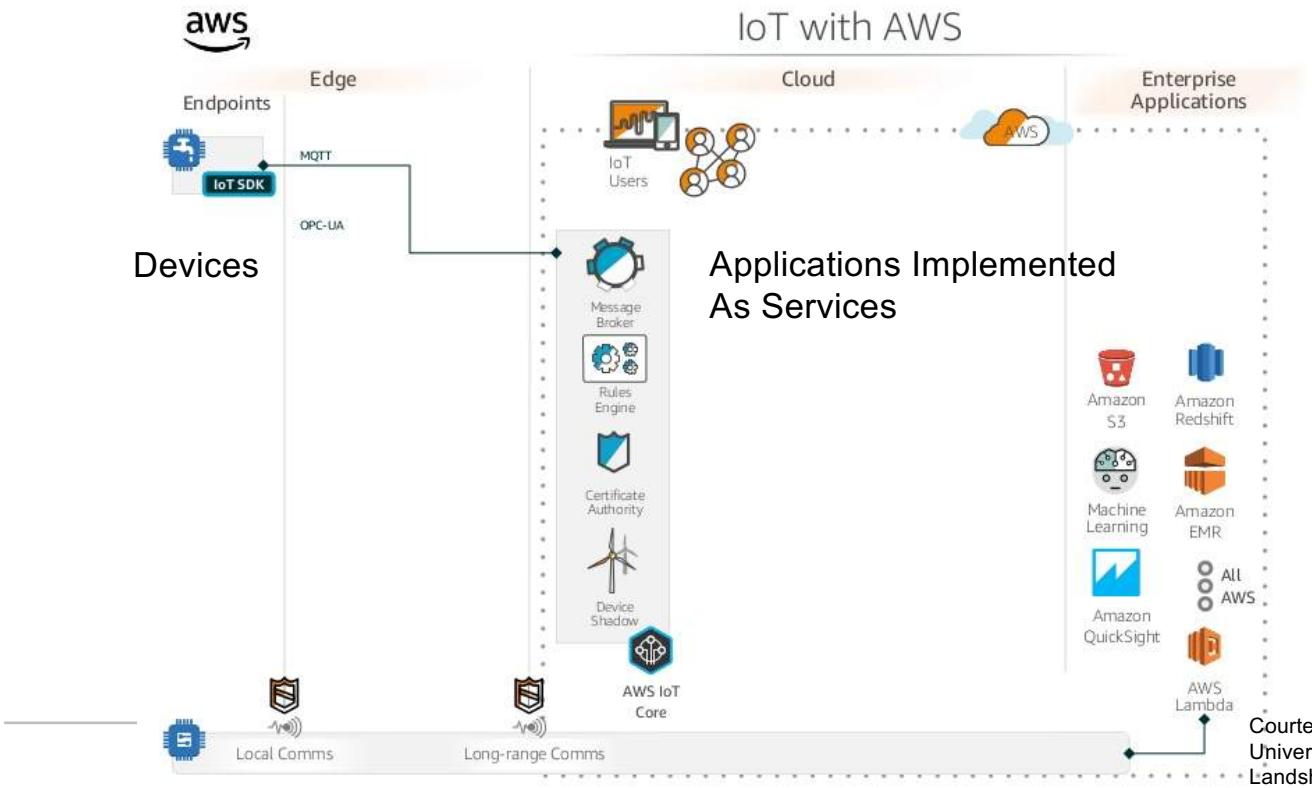


Microsoft Azure

Google Cloud Platform

UCSB

Today: Devices are Cloud “Users” and Applications are Services



Courtesy Dr. Markus Mock, University of Applied Sciences, Landshut, Germany

UCSB

Lessons from Cloud Computing

- Services and service-based computing is
  - Robust
  - Maintainable at scale
  - Portable
  - Efficient
  - Scalable
- We create services to implement multi-function implementations that manipulate state via well-defined APIs

Ergo, for IoT, Devices Should Run Services

- Devices are increasingly multi-function and stateful (at least temporarily)
- Service composition is well-understood from a software engineering perspective
- Challenges to Developing Devices-as-Services
  - Must be efficient
    - Power, speed, space
  - Must be universal and multi-scale
    - Device platforms are heterogeneous in architecture and capacity dimensions
  - Must be distributed
    - IoT requires a great deal of “in the field” computing

Start with Functions-as-a-Service

Google Cloud Functions

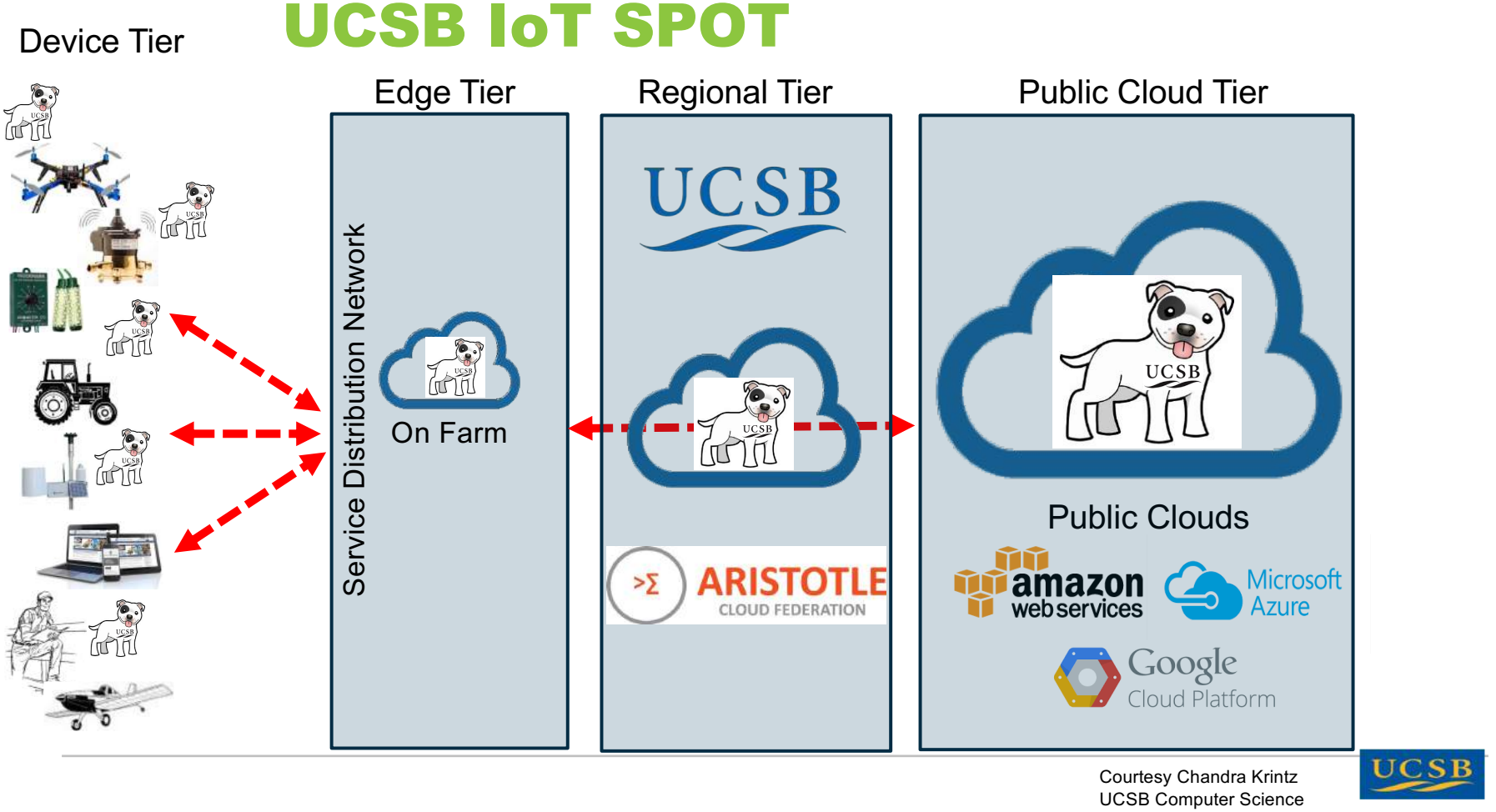
- Also called “FaaS” or “serverless” computing
  - AWS Lambda, Google Functions, Azure Functions
- Developer writes “stateless” functions in high-level language
- Event-driven programming model
  - Functions are only triggered when a specific event occurs
- Functions are isolated (usually via Linux containers)
- In a cloud, FaaS allows inexpensive and easy development and deployment of autoscaling web services

CSPOT

- **S**erverless **P**latform **O**f **T**hings (in **C**)
- Multi-scale, distributed FaaS
  - Run on microcontrollers, small Linux (Raspberry Pi, Intel NUC), Campus Clouds (NSF Aristotle), public clouds (AWS, Azure, Google, IBM)
  - CSPOT program ports (source code level) without modification between scales
- CSPOT includes (first distributed FaaS)
  - Append-only storage abstraction (distributed data durability, eventual consistency)
  - Log-based runtime that tracks causal dependencies (debugging)
- ***CSPOT is a way to write multi-scale network-facing services***
- Open Source
  - https://github.com/MAYHEM-Lab/cspot.git
  - Join us! Contribute!

Flood Prevention

UCSB IoT SPOT



Courtesy Chandra Krintz
UCSB Computer Science

Portable and Power Efficient

- Dispatch times for clock read FaaS Handlers
- CSPOT is 1 to 2 orders of magnitude faster than AWS Lambda
- Power being proportional to latency

System	Mean (ms)	Stdev (ms)	95% (ms)
Esp8266 ucontroller	38	0.15	40
Raspberry Pi (ARM)	37	6.8	48
Intel NUC	4.0	0.63	4.9
UCSB Cloud	5.0	1.6	7.0
AWS SPOT EC2 in C	5.0	0.96	6.6
AWS SPOT EC2 in Python	18	3.1	23
AWS Lambda	*253*	*90*	*584*

UCSB

Behold! The IoT Cloud

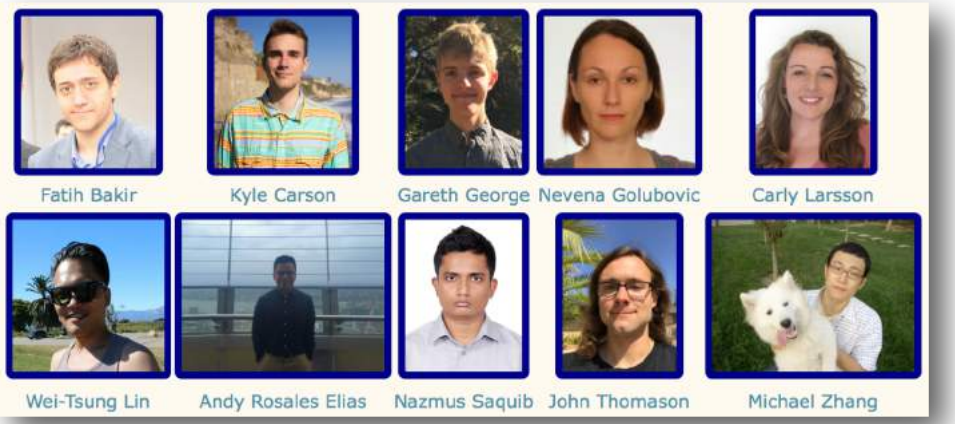
UCSB

Thanks!

- Collaborators: UCSB, UCSB IEE, LREC, CalPoly, Fresno State, Powwow Energy, Sedgwick Reserve, Private Growers

- Support: Google, Intel, IBM Research, Microsoft Research, NSF, NIH, California Energy Commission

Students:



UCSB RACELab

The Lab for Research on Adaptive Computing Environments
Computer Science Department, Harold Frank Hall (E-5), Santa Barbara, CA

rich@cs.ucsb.edu, ckrintz@cs.ucsb.edu