MASSIVIZING COMPUTER SYSTEMS

= MAKING COMPUTER SYSTEMS SCALABLE, RELIABLE, PERFORMANT, ETC., YET ABLE TO FORM AN EFFICIENT ECOSYSTEM

THE SCIENCE OF DISTRIBUTED (ECO)SYSTEMS

http://atlarge.science

Co-sponsored by:
NWO Netherlands Organisation for Scientific Research
STW connecting innovators

VRIJE UNIVERSITEIT AMSTERDAM

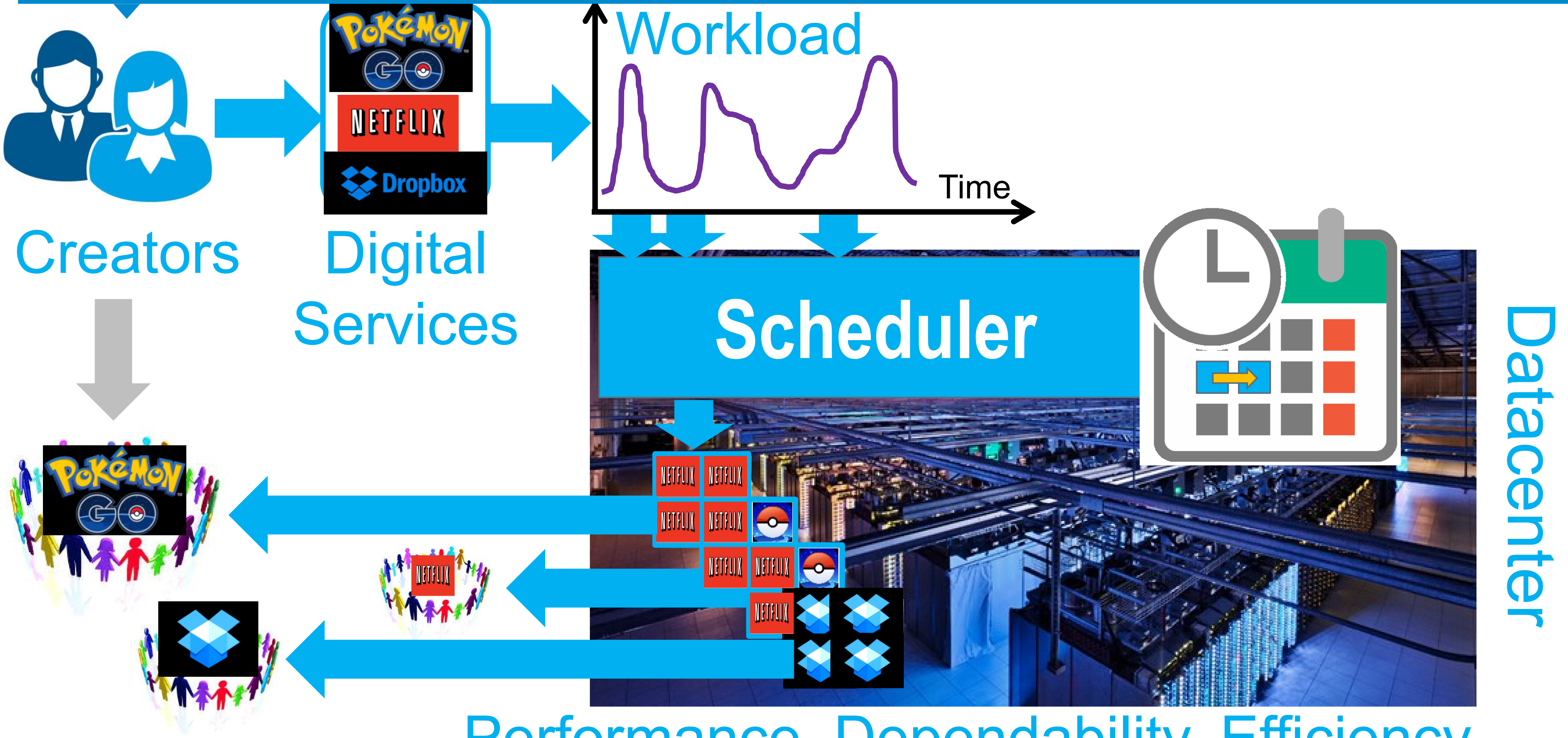
@AIosup

Prof. dr. ir. Alexandru Iosup

THIS IS THE GOLDEN AGE OF DISTRIBUTED ECOSYSTEMS

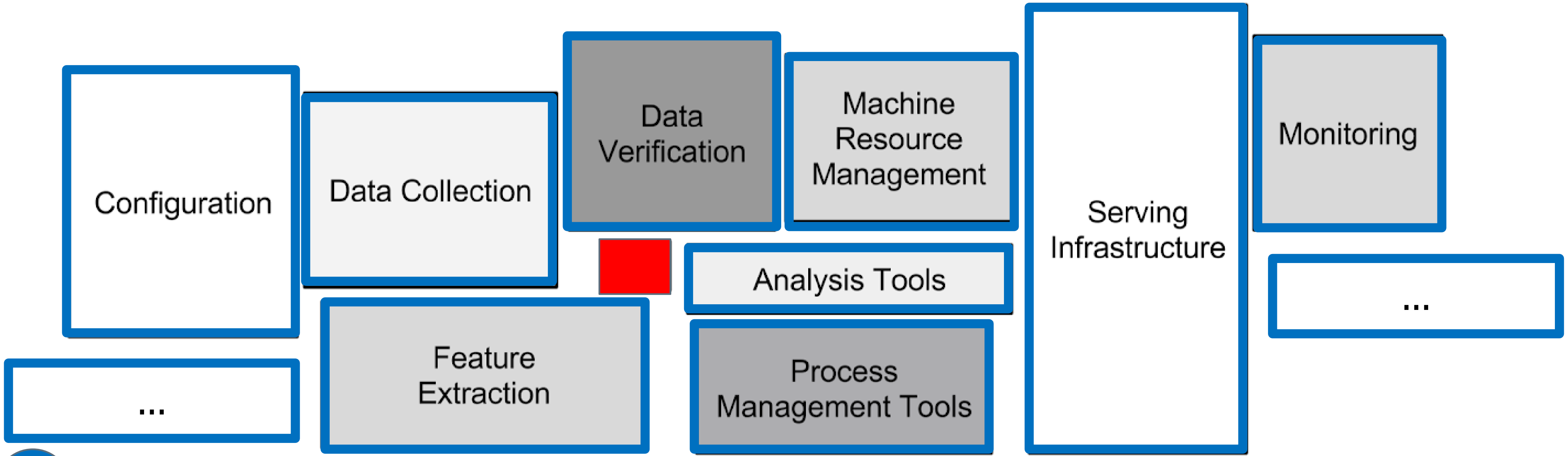
THIS IS THE GOLDEN AGE OF DISTRIBUTED COMPUTER SYSTEMS

THE CURRENT TECHNOLOGY STACK: DATACENTER, SCHEDULER



Performance, Dependability, Efficiency

WHY THIS COURSE? EVEN THE HIGHLY PROMOTED MACHINE LEARNING IS BY AND LARGE ABOUT COMPUTER SYSTEMS



1 Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small [red] box in the middle. The required surrounding infrastructure is vast and complex.

2 Box size [box] ~ effort and complexity of the task.

Hidden technical debt in Machine Learning Systems - Google, NIPS '15

Agenda

1. Why this course?
2. **A Theory of Distributed Systems**
3. A History of Distributed Systems
4. The Science of Distributed Systems
5. The Virtuous Cycle: a Science-Design-Engineering Example
6. Take-Home Message

DEF: What is a Distributed Ecosystem?

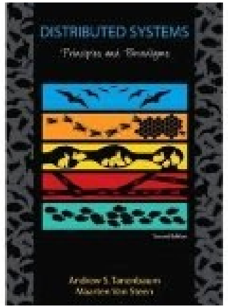
1. Set of 2+ constituents, often heterogeneous
2. Each constituent is a system or an ecosystem (recursively)
3. Constituents are autonomous, cooperative or in competition
4. Ecosystem structure and organization ensure responsibility
    1. Completing functions
    2. Providing desirable non-functional properties
    3. Fulfilling agreements of 1+2 for clients, with the clients in the loop
5. Long and short-term dynamics occur in the ecosystem

Iosup et al., Massivizing Computer Systems, ICDCS 2018. [Online]

DEF: What is a Distributed System?

"You know you have a distributed system when the crash of a computer you've never heard of stops you from getting any work done."
- **Leslie Lamport** in Security Engineering, Ch.6

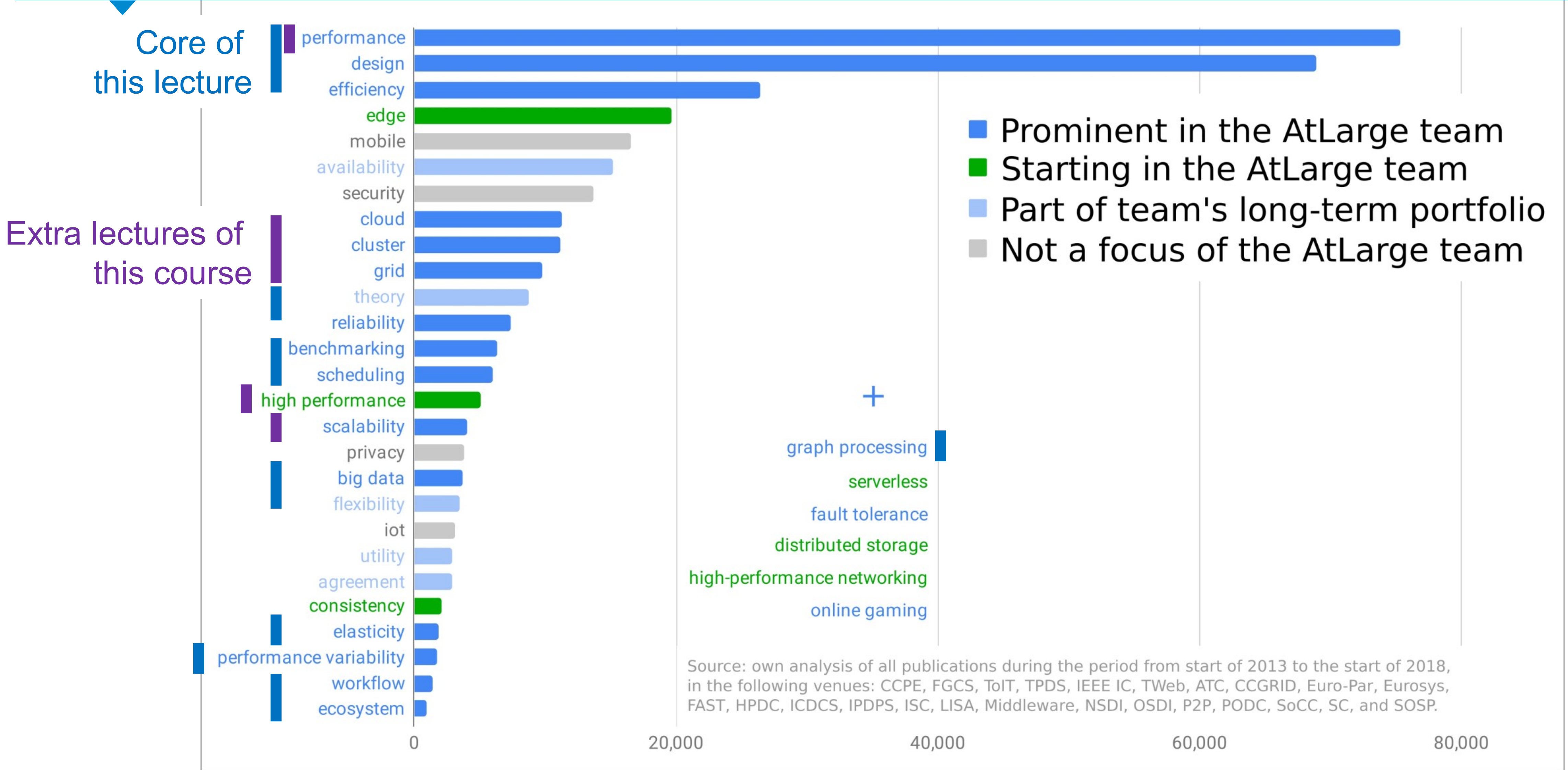
*"A collection of **autonomous computing elements** that appears to its users as a single coherent system* - **Steen and Tanenbaum** in Distributed Systems: Principles and Paradigms, 3rd Edition, 2017



"an application that executes a collection of protocols to coordinate the actions of multiple processes on a network, such that all components cooperate together to perform a single or small set of related tasks."- Google University, Introduction to DS Design
http://www.hpcs.cs.tsukuba.ac.jp/~tatebe/lecture/h23/dsys/dsd-tutorial.html

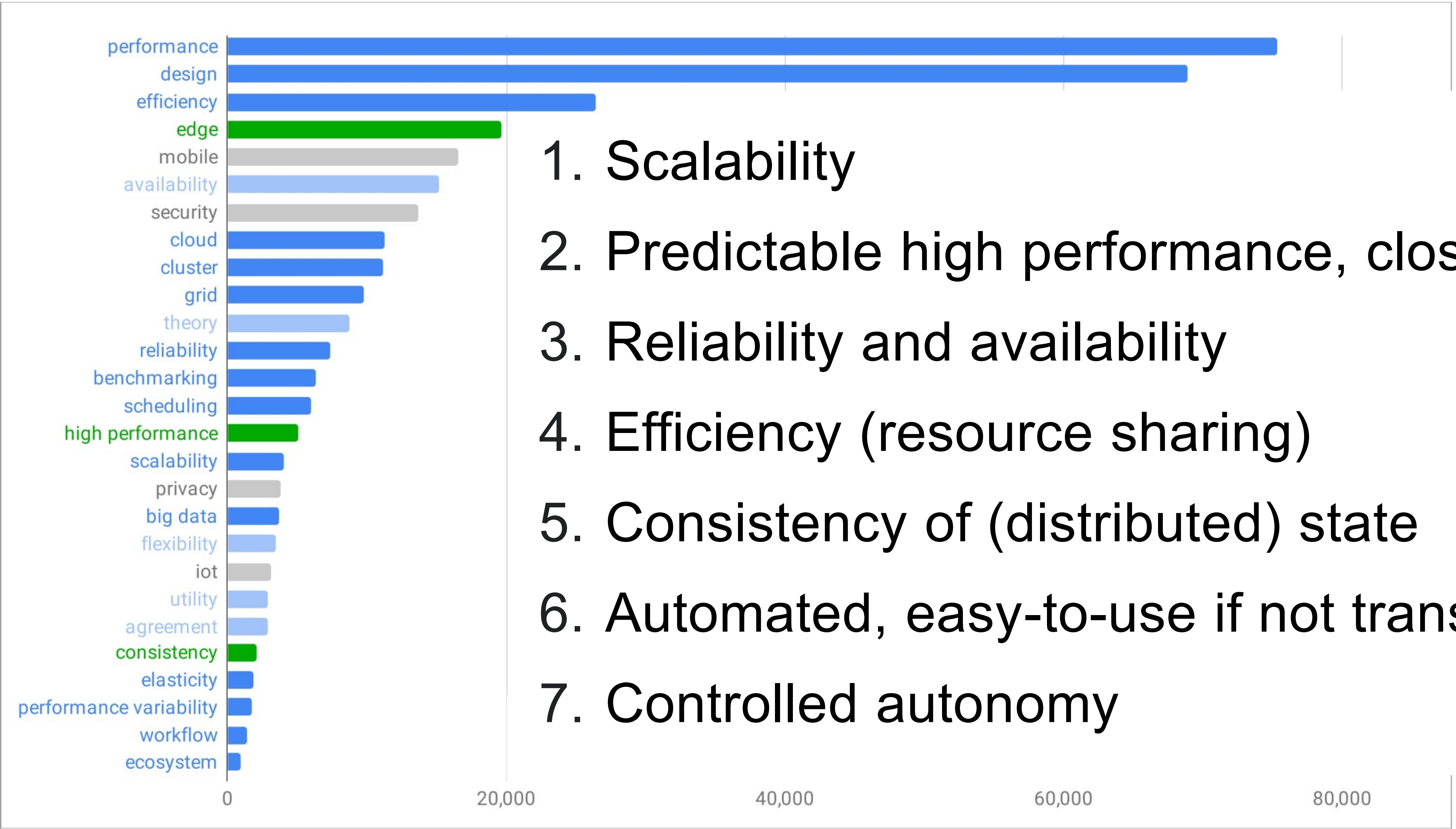
DISTRIBUTED SYSTEMS
WHAT YOU NEED TO KNOW (END-2010S)

http://atlarge.science



What Should You Expect from a Distributed Ecosystem?

Main Characteristics of Distributed (Eco)Systems



1. Scalability
2. Predictable high performance, close-to-users
3. Reliability and availability
4. Efficiency (resource sharing)
5. Consistency of (distributed) state
6. Automated, easy-to-use if not transparent
7. Controlled autonomy

The Core Idea through An Example
BitTorrent: A Distributed System

Q: Autonomy? Cooperation? Communication?

Q: Does this system scale? Why? How?

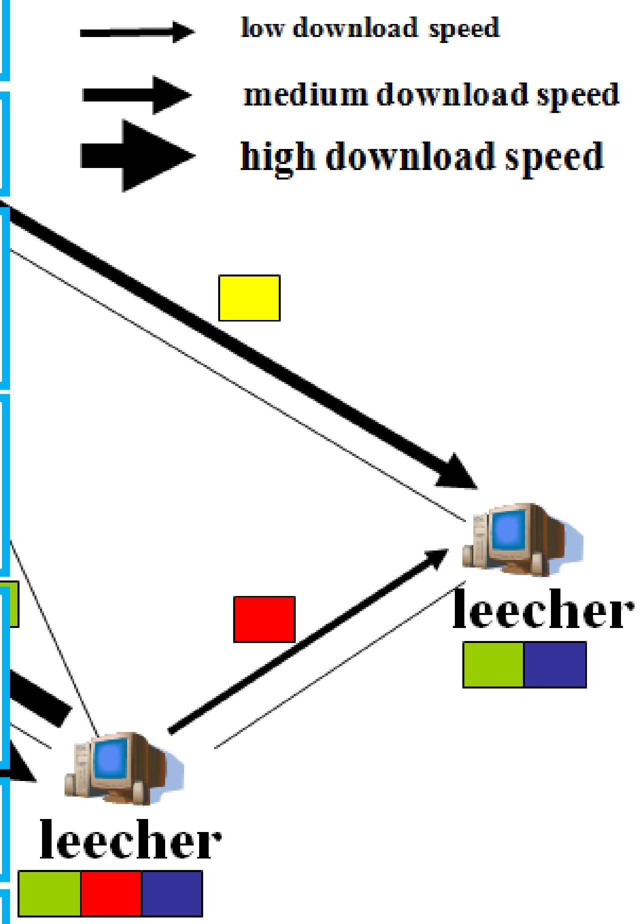
Q: What is the structure of this system?
What is the state of each node? How do they synchronize?

Q: How does the performance of this system change with the increase in the number of users?

Q: When is this system available?
What does it do to increase its reliability?

Q: Is this system efficient?

Q: Which parts of this system need consistency? Achieved?



d.

Distributed Variants

- Most grid computing
- Most cloud computing
- Peer-to-Peer computing
- Most Big Data processing (MapReduce/Hadoop, Pregel/Giraph, Spark, etc.)
- Cluster computing
- Some High-Performance Computing

Distributed Q: Cluster of GPUs? Parallel Computing

Q: Cluster computing?	Q: GPU processing?
Multiple tasks, one job or multiple jobs	Multiple tasks, one job
Throughput or Speed-up	Speed-up
Horizontal scaling	Vertical scaling
Infrequent communication	Frequent communication
Synchronized execution	Simultaneous execution
Heterogeneous hardware	Homogenous hardware
Multiple owners with mutual interests	Single owner

Agenda

1. Why this course?
2. A Theory of Distributed Systems
3. **A History of Distributed Systems**
4. The Science of Distributed Systems
5. The Virtuous Cycle: a Science-Design-Engineering Example
6. Take-Home Message

History of Distributed Systems and Ecosystems

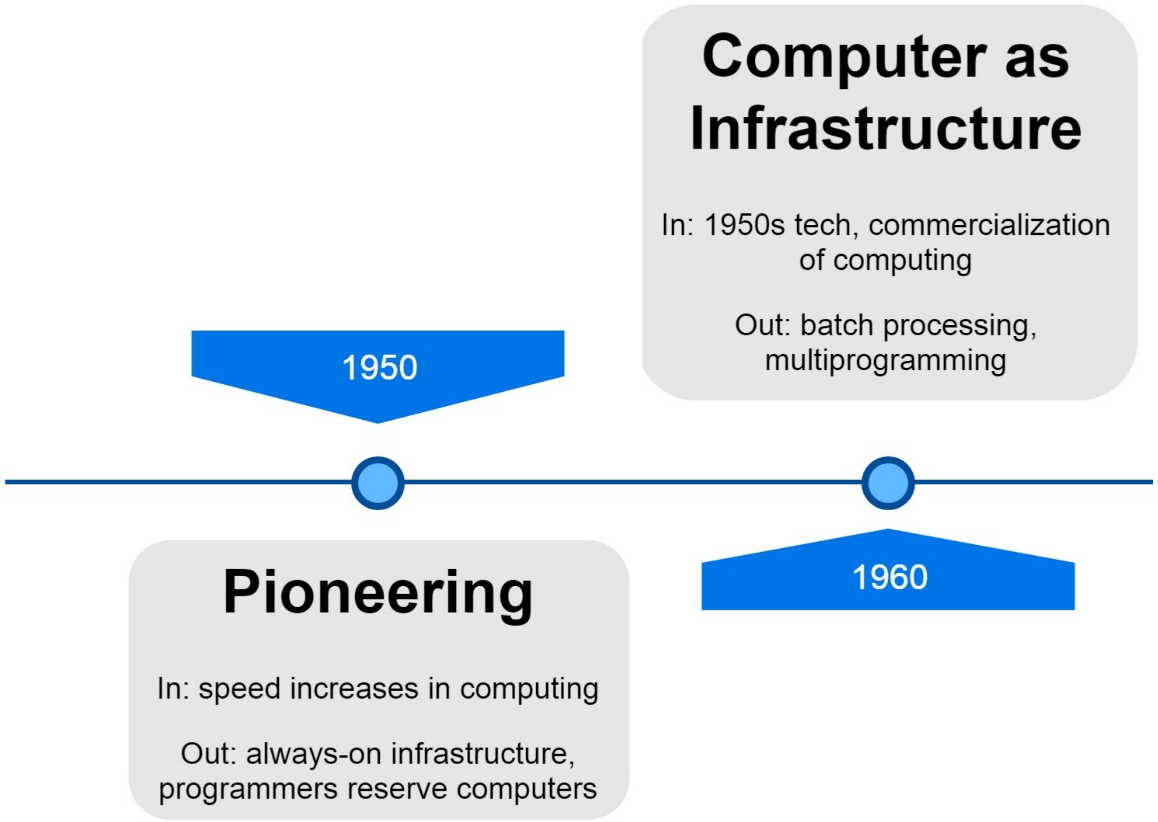
1950

Pioneering

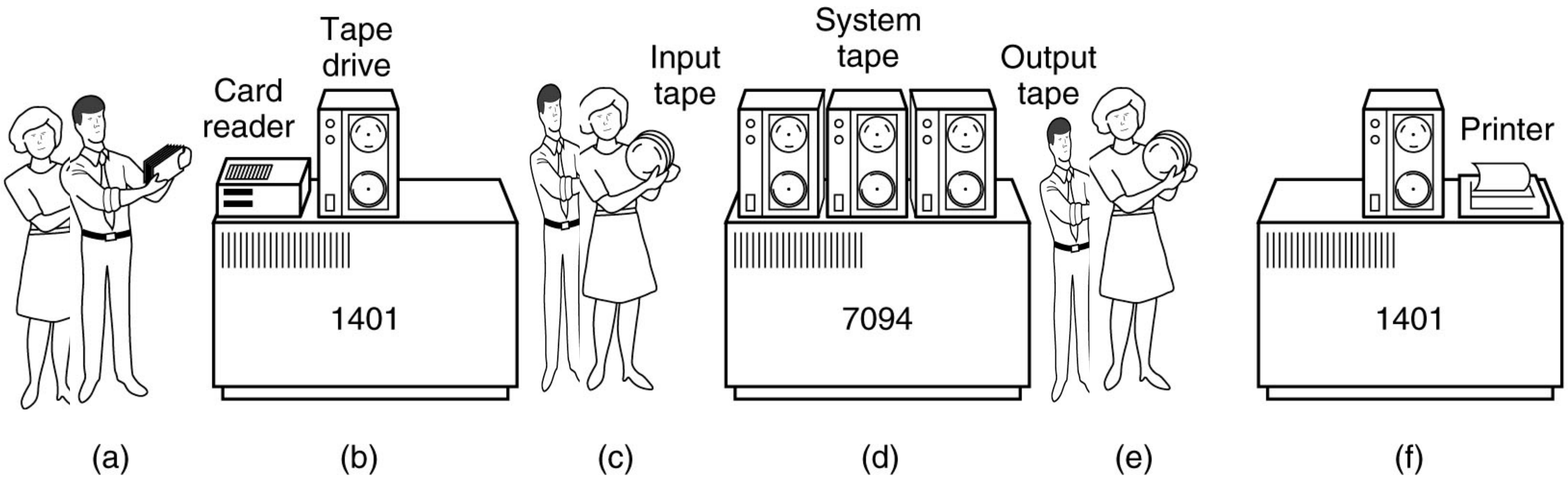
In: speed increases in computing

Out: always-on infrastructure, programmers reserve computers

History of Distributed Systems and Ecosystems



Computer Time Expensive, Human Labor Cheap
→ Batch Processing, Multiprogramming (1960s)

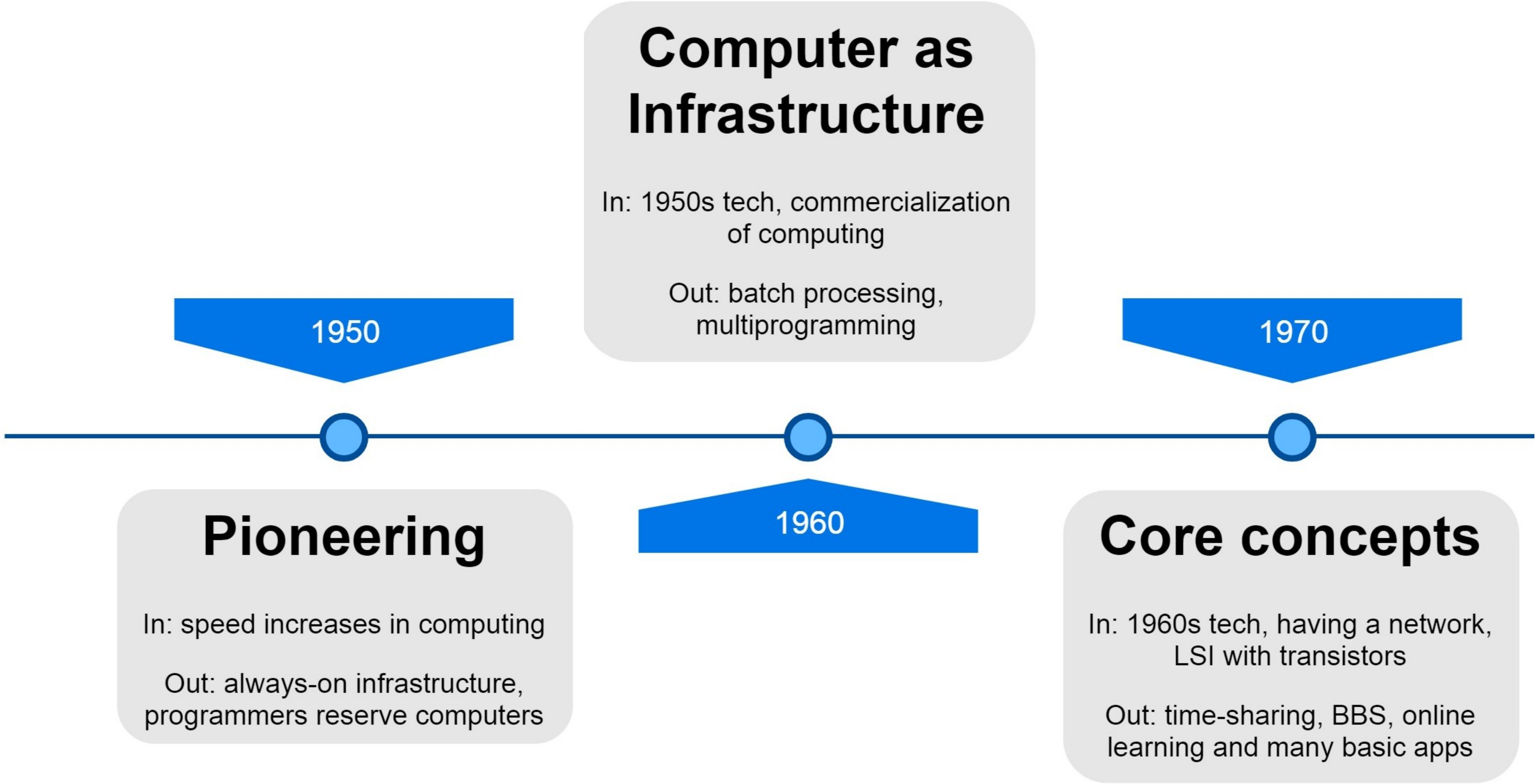


Programmers Reader Operator **Computer** Operator Printer

(Corbato's CTSS, at MIT)

https://multicians.org/thvv/7094.html

History of Distributed Systems and Ecosystems



Early Networks, Early Basic Apps (1970s)

- PLATO
  - Early learning management system
  - Distributed, thousands of deployments
- Bulletin Board Systems (BBS)
  - Seen as systems for information dissemination
  - Early business applications, esp. financial

History of Distributed Systems and Ecosystems

1950

Pioneering

In: speed increases in computing

Out: always-on infrastructure, programmers reserve computers

1960

Computer as Infrastructure

In: 1950s tech, commercialization of computing

Out: batch processing, multiprogramming

1970

Core concepts

In: 1960s tech, having a network, LSI with transistors

Out: time-sharing, BBS, online learning and many basic apps

1980

For Everyone

In: 1970s tech, commercialization of DistribSys, personal computer, commodity LAN

Out: point-to-point apps, distributed file systems, lots of apps, also the sneakernet, plus many distributed algorithms and computing models

The IBM Personal Computer

- Released in the 1980s
  - The blueprint for today's PCs
  - Changed the market
- Open standards and friendliness to third-party hardware and software developers

Q: How did they transfer data between **PCs?**

- Data transfers between PCs
  - Commodity LANs in enterprises
  - Sneakernet for cosumers

History of Distributed Systems and Ecosystems

1990

The Internet

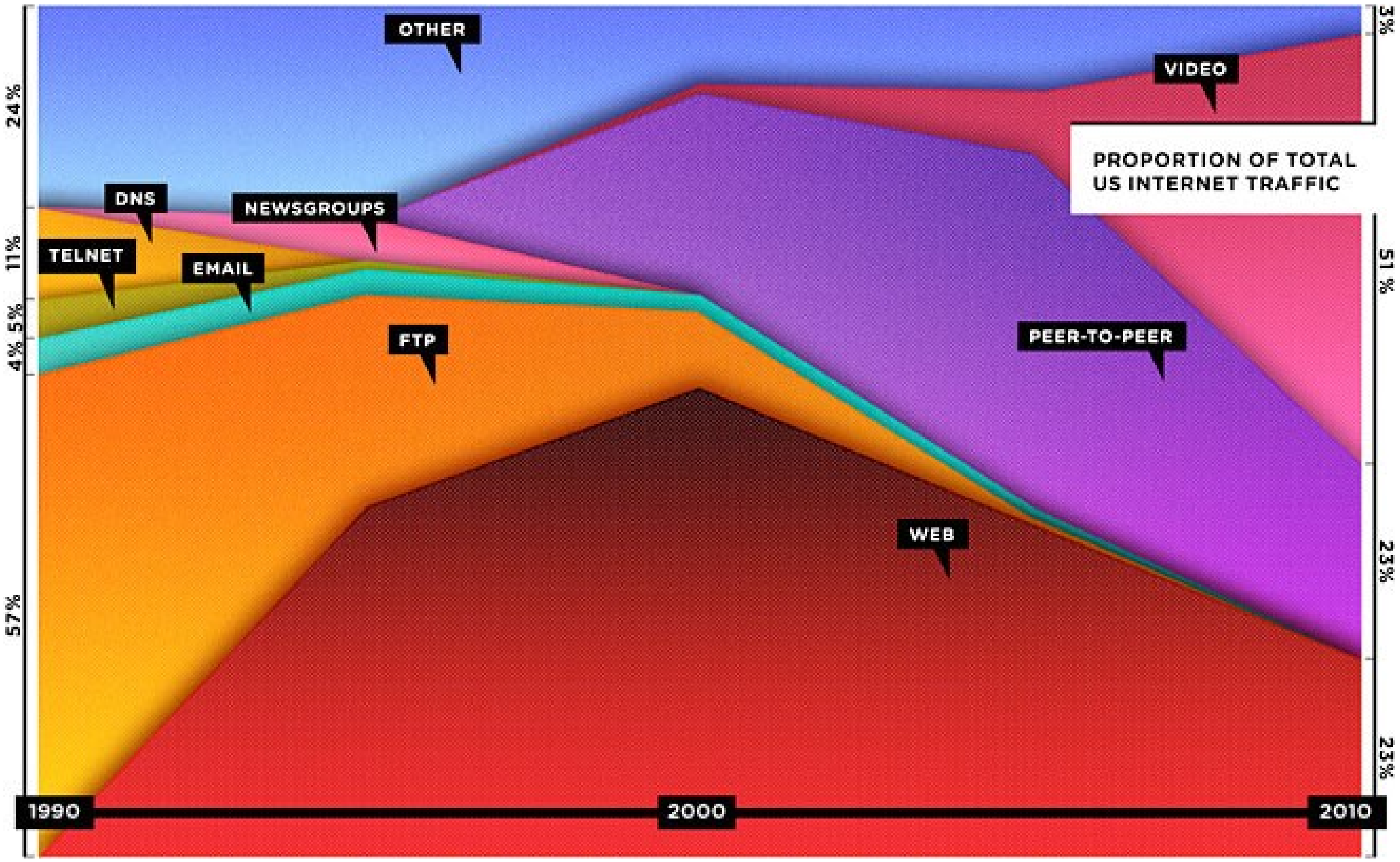
In: 1980s tech, standardization on hardware, commercialization of distributed services, WANs, DNS

Out: WWW, browsers, middleware, supercomputing, grid computing, online gaming, early mobile

Multi-computing

Internet-Based Applications

- Metcalfe's Law: usefulness of a network ~ n^2, n objects/users



digital movies >> DVD (2015)

History of Distributed Systems and Ecosystems

eCommerce

In: 1990s tech, commercialization of the Internet, ADSL, cell phones

Out: online shopping, MMO games, music streaming, file sharing, P2P, Datacenters, e-Science, sensors, mobile, online social networks

1990

2000

The Internet

In: 1980s tech, standardization on hardware, commercialization of distributed services, WANs, DNS

Out: WWW, browsers, middleware, supercomputing, grid computing, online gaming, early mobile

Consumer: A Computer In Your Pocket (or Hand)

- Not tech, apps + mobility
- Internet everywhere (?)
- PC killers (?!)
- iPhone and relatives (2007—)
  - Initially music device++
- iPad and relatives (2010—)
  - Small format, high res
- 2013: smart- >> cell-phone

Source: http://www.imore.com/history-iphone-2g + http://www.imore.com/history-ipad-2010

History of Distributed Systems and Ecosystems

1990

The Internet

In: 1980s tech, standardization on hardware, commercialization of distributed services, WANs, DNS

Out: WWW, browsers, middleware, supercomputing, grid computing, online gaming, early mobile

2000

eCommerce

In: 1990s tech, commercialization of the Internet, ADSL, cell phones

Out: online shopping, MMO games, music streaming, file sharing, P2P, Datacenters, e-Science, sensors, mobile, online social networks

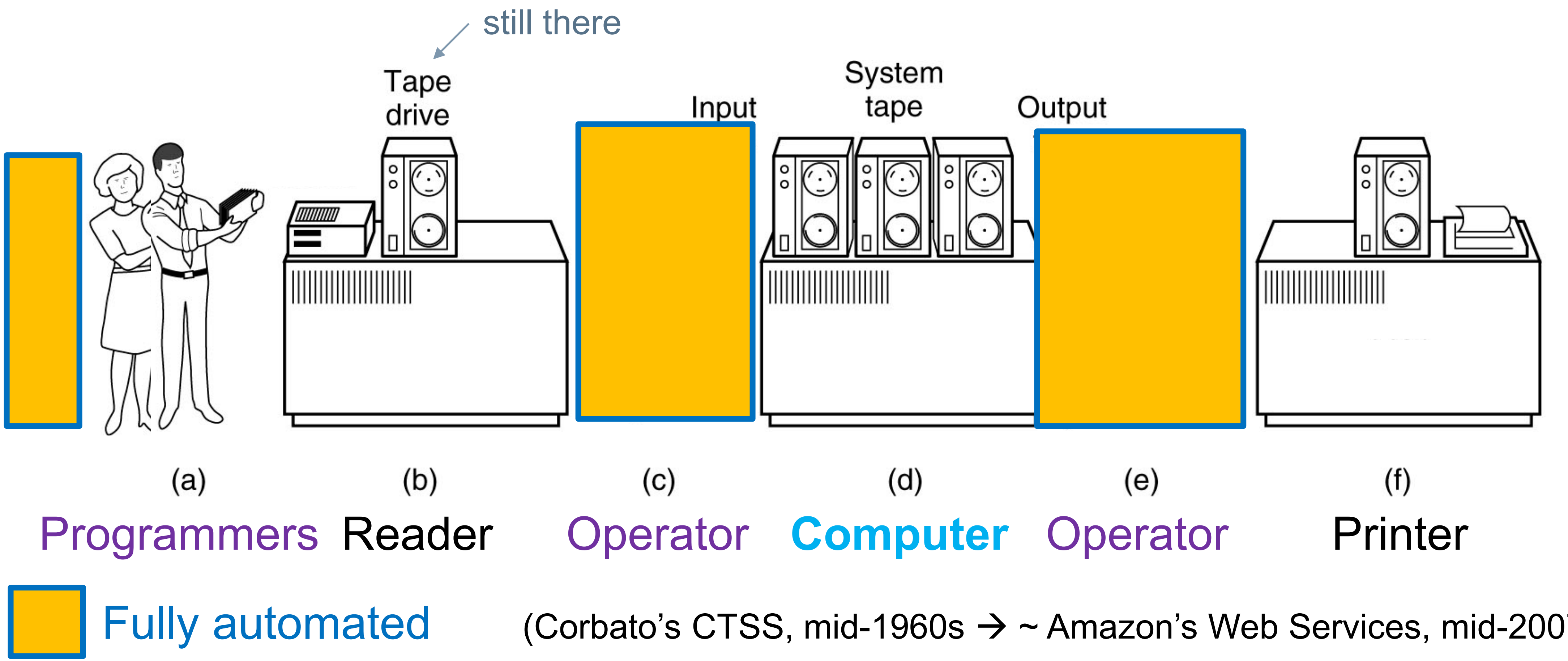
2010

Digital Services

In: 2000s tech, commercialization of online services, payment revolution, home broadband

Out: e-everything, cloud-native, mobile-first, big data, video streaming, smartphones, basic ML, privacy troubles

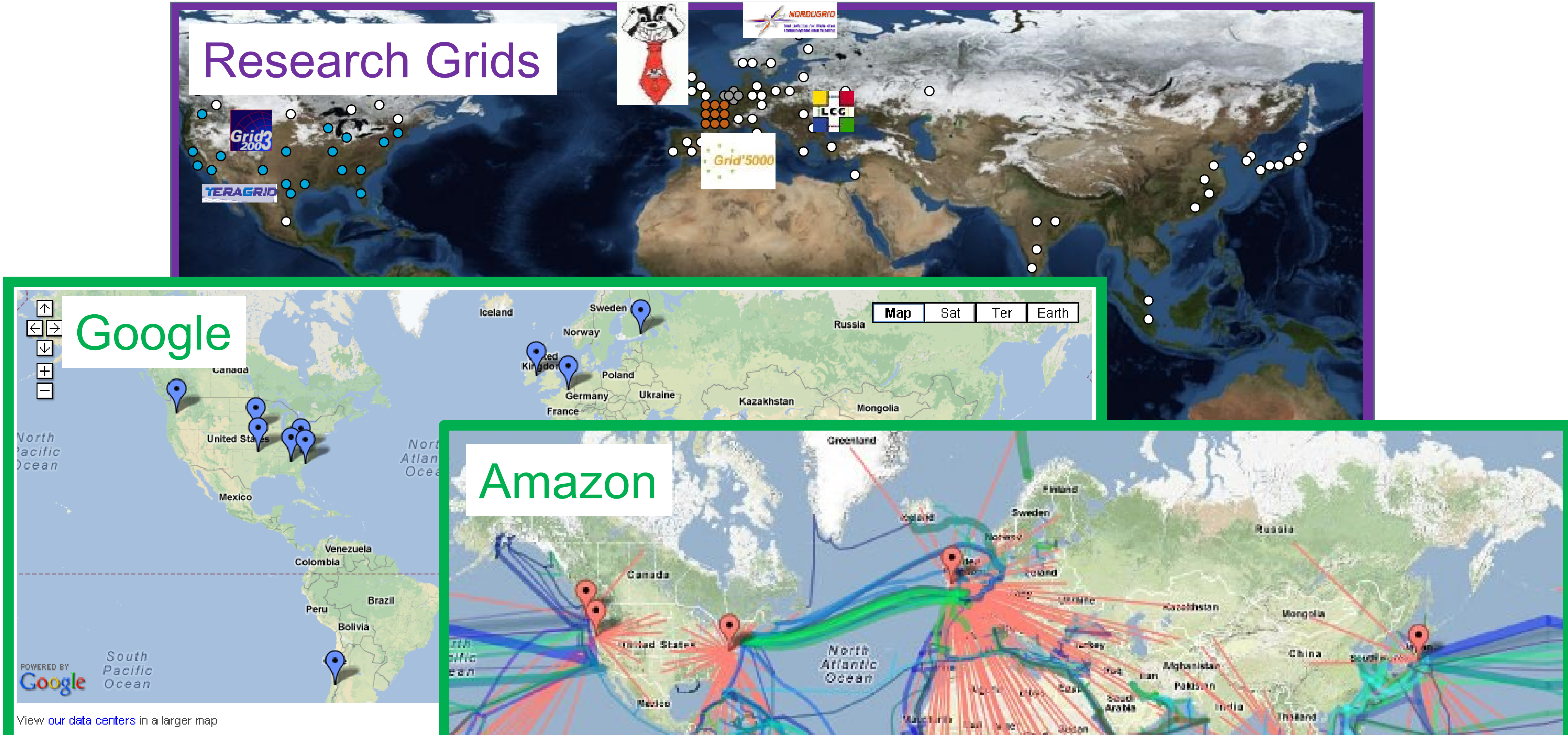
Computer Time Cheap, Human Labor Expensive
→ Batch Processing, Cloud Computing (2010s)



(Corbato's CTSS, mid-1960s → ~ Amazon's Web Services, mid-2007s)

https://multicians.org/thvv/7094.html

Grids ('00s) and Cloud/Datacenter Computing ('10s)



History of Distributed Systems and Ecosystems

1990

The Internet

In: 1980s tech, standardization on hardware, commercialization of distributed services, WANs, DNS

Out: WWW, browsers, middleware, supercomputing, grid computing, online gaming, early mobile

2000

eCommerce

In: 1990s tech, commercialization of the Internet, ADSL, cell phones

Out: online shopping, MMO games, music streaming, file sharing, P2P, Datacenters, e-Science, sensors, mobile, online social networks

2010

Digital Services

In: 2000s tech, commercialization of online services, payment revolution, home broadband

Out: e-everything, cloud-native, mobile-first, big data, video streaming, smartphones, basic ML, privacy troubles

Future

Ecosystems for everyone?

In: 2010s tech, commercialization of machine learning

Out: digital rights, future of banking, bioinfo, wellbeing, gaming, etc

Agenda

1. Why this course?
2. A Theory of Distributed Systems
3. A History of Distributed Systems
4. **The Science of Distributed Systems**
5. The Virtuous Cycle: a Science-Design-Engineering Example
6. Take-Home Message

Principles of Distributed Systems

P0: The Golden Age.

P1: Super-distribution.

P2: Programming Model - System Architecture interaction.

P3: Resource Management and Scheduling at the core.

P4: Non-functional requirements as core concern.

P5: Inherent functional requirements.

P6: Design for massive scale.

P7: Pervasive phenomena.

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE

GALILEO GALILEI, 1608-9, 3-8X TELESCOPE

MERELY AN INSTRUMENT?

FUNDAMENTAL SCIENCE?

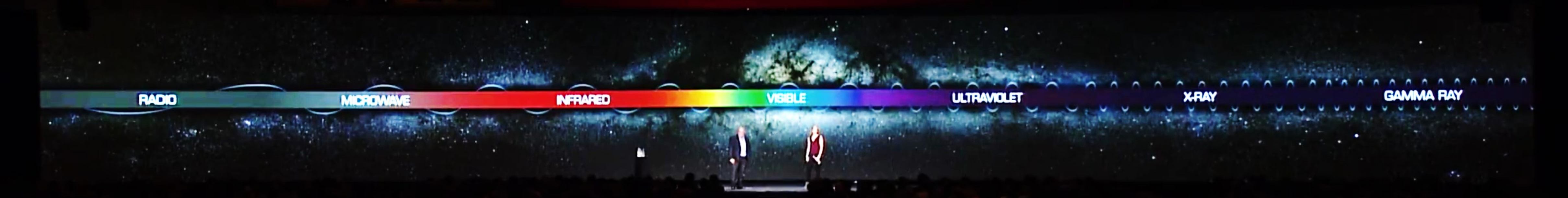
Garney. The Inquisition’s Semicolon: Punctuation, Translation, and Science in the 1616 Condemnation of the Copernican System, ArXiv document 1402.6168. [Online]

Phil Diamond and Rosie Bolton, Life, the Universe & Computing: The story of the SKA Telescope, SC17 Keynote. [Online]

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE

400 YEARS LATER, SYSTEMATIC SURVEY



Radio Microwave Infrared Visible Ultraviolet X-Ray Gamma Ray

Phil Diamond and Rosie Bolton, Life, the Universe & Computing: The story of the SKA Telescope, SC17 Keynote. [Online]

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE

GEORGE SMOOT
NOBEL PRIZE 2006

SKA FUNDING: 500+ FTE, EUR 1.5B

James Cordes, The Square Kilometer Array, Project Description, 2009 [Online]
The Square Kilometer Array Factsheet, How much will it cost?, 2012 [Online]
Phil Diamond and Rosie Bolton, Life, the Universe & Computing: The story of the SKA Telescope, SC17 Keynote. [Online]

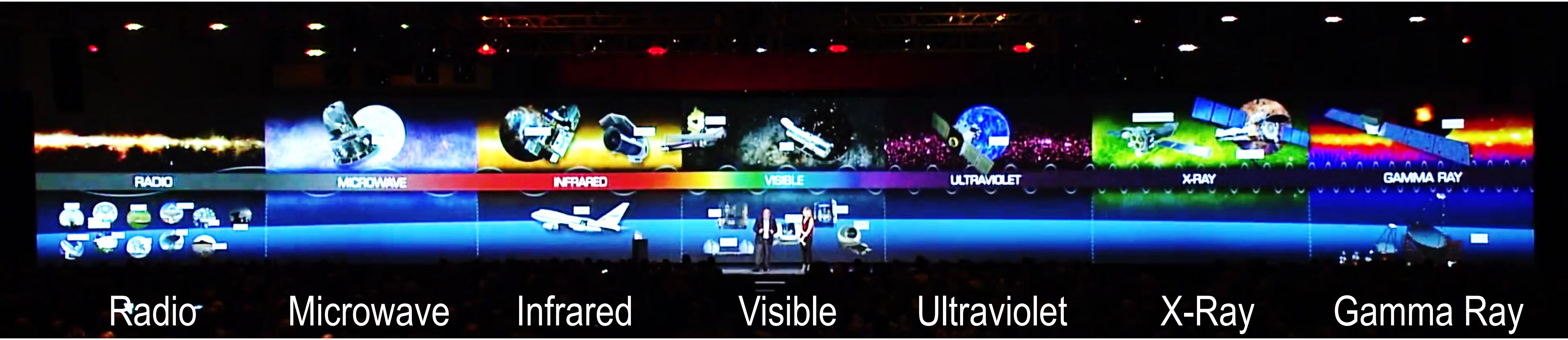
MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE, PHYSICAL AND DIGITAL

Radio	Microwave	Infrared	Visible	Ultraviolet	X-Ray	Gamma Ray
Cloud, Grid, Edge, Fog, etc.	Big Data	Sci.&Eng. Apps	Consumer Apps	Enterprise Apps	Systems, Ecosystems	Performance, Security, etc.

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE, PHYSICAL AND DIGITAL



Cloud, Grid, Edge, Fog, etc. | One aspect: BigData, P2P | Sci.&Eng. Apps+Sys. | Consumer Apps+Sys. | Enterprise Sys. | Systems, Ecosystems | Performance, Availability, etc.

[Iosup et al. FGCS'08]

P2P T.A.

[Zhang et al. CoNext'10]

BoTs, Groups, Workflows

[Iosup et al. IEEE IC'11]

Game Trace Archive

[Guo et al. NETGAMES'12]

Business -Critical

[Shen et al. CCGRID'15]

[Ghiț et al. CCGRID'14]

[Iosup et al. CCGRID'10]

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE, PHYSICAL AND DIGITAL

BUT … WHY WOULD YOU NEED TO UNCOVER AN ARTIFICIAL UNIVERSE?! YOU BUILT IT!

Cloud, Grid, Edge, Fog, etc.

[Iosup et al. FGCS'08]

One aspect: BigData, P2P

P2P T.A.

[Zhang et al. CoNext'10]

Sci.&Eng. Apps+Sys.

BoTs, Groups, Workflows

[Iosup et al. IEEE IC'11]

Consumer Apps+Sys.

Game Trace Archive

[Guo et al. NETGAMES'12]

Enterprise Sys.

Business -Critical

[Shen et al. CCGRID'15]

Systems, Ecosystems

[Ghiț et al. CCGRID'14]

Performance, Availability, etc.

[Iosup et al. CCGRID'10]

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE, PHYSICAL AND DIGITAL

FOUND MANY UNFORESEEN PHENOMENA: INTERACTION, ADAPTATION, EXAPTATION, …

…	…	…	…	…
BOTS, NOT PARALLEL JOBS	GROUPS NOT RARE, DOMINANT	COMMUNITY FORMATION	SYSTEMIC VARIABILITY	CORRELATED, NOT IID FAILURES

Cloud, Grid, Edge, Fog, etc.

One aspect: BigData, P2P

Sci.&Eng. Apps+Sys.

Consumer Apps+Sys.

Enterprise Sys.

Systems, Ecosystems

Performance, Availability, etc.

[Iosup et al. FGCS'08]

P2P T.A.

[Zhang et al. CoNext'10]

BoTs, Groups, Workflows

[Iosup et al. IEEE IC'11]

Game Trace Archive

[Guo et al. NETGAMES'12]

Business -Critical

[Shen et al. CCGRID'15]

[Ghiț et al. CCGRID'14]

[Iosup et al. CCGRID'10]

MEANINGFUL DISCOVERY

UNCOVERING THE MYSTERIES OF OUR UNIVERSE, PHYSICAL AND DIGITAL

FOUND MANY UNFORESEEN PHENOMENA: INTERACTION, ADAPTATION, EXAPTATION, …

BUT … IS THERE A SYSTEMATIC WAY TO APPROACH THESE PHENOMENA?

BOTS, NOT PARALLEL JOBS	GROUPS NOT RARE, DOMINANT	COMMUNITY FORMATION	SYSTEMIC VARIABILITY	CORRELATED, NOT IID FAILURES

Cloud, Grid, Edge, Fog, etc. One aspect: BigData, P2P Sci.&Eng. Apps+Sys. Consumer Apps+Sys. Enterprise Sys. Systems, Ecosystems Performance, Availability, etc.

[Iosup et al. FGCS'08]

P2P T.A.

[Zhang et al. CoNext'10]

BoTs, Groups, Workflows

[Iosup et al. IEEE IC'11]

Game Trace Archive

[Guo et al. NETGAMES'12]

Business -Critical

[Shen et al. CCGRID'15]

[Ghiț et al. CCGRID'14]

[Iosup et al. CCGRID'10]

MEANINGFUL DISCOVERY

BUT … IS THERE A SYSTEMATIC WAY TO APPROACH THESE PHENOMENA?

FUNDING: > 3B USD

- The Human Genome Project:
  - Physical map covering >90% human genome
  - Sequence data made available open-access
- Big Science:
  - Took >10 years to complete
  - Led by US, work by 20 groups in CN, DE, FR, JP, UK, US
- Big impact:
  - Decrease cost of sequencing
  - Facilitate biomedical research

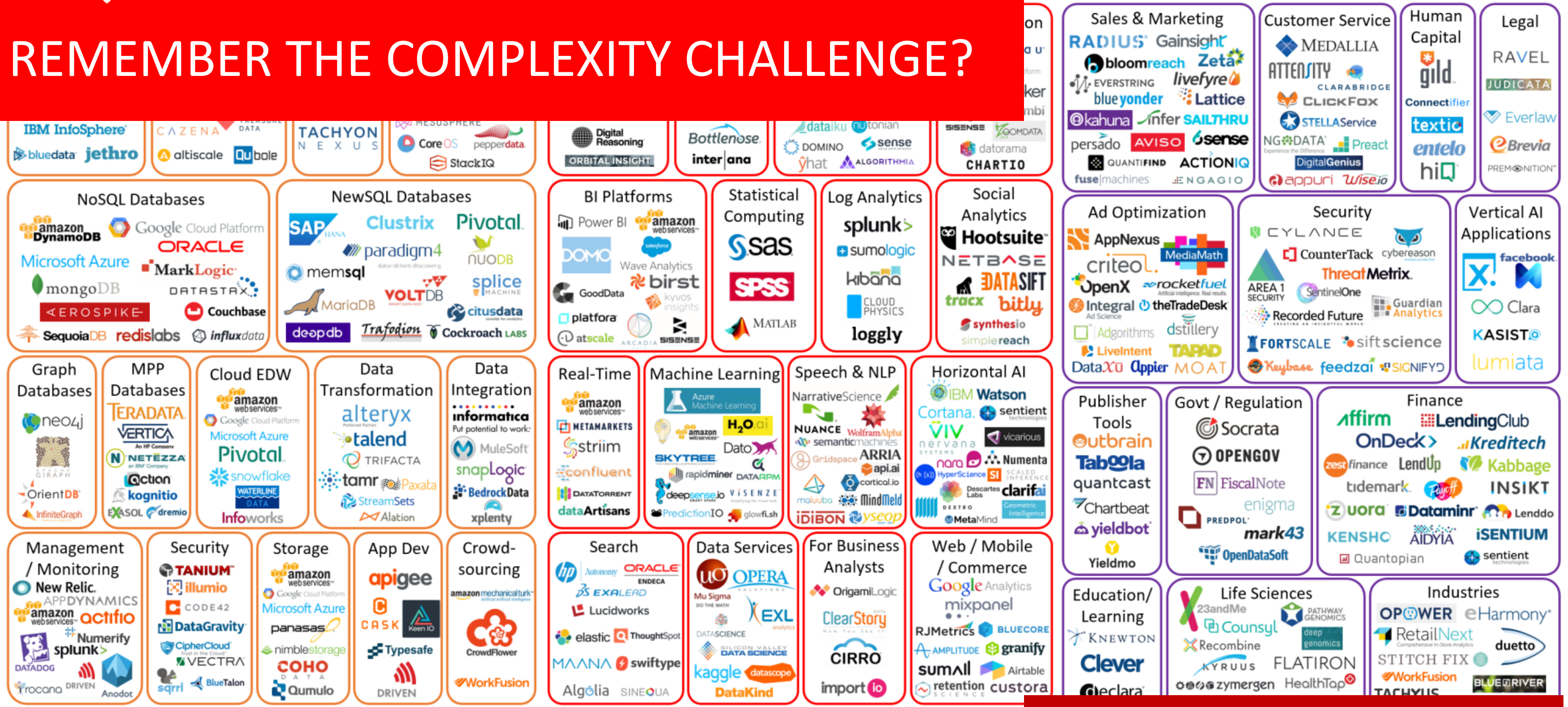
International Human Genome Sequencing Consortium, Initial sequencing and analysis of the human genome, Nature 409, Feb 2011. [Online]

Julie Gould, The Impact of the Human Genome Project, Naturejobs blog, 2015. [Online]

MEANINGFUL DISCOVERY

BUT … IS THERE A SYSTEMATIC WAY TO APPROACH THESE PHENOMENA?

REMEMBER THE COMPLEXITY CHALLENGE?



<<1% OF BIG DATA
BY MATT TURK (2017)

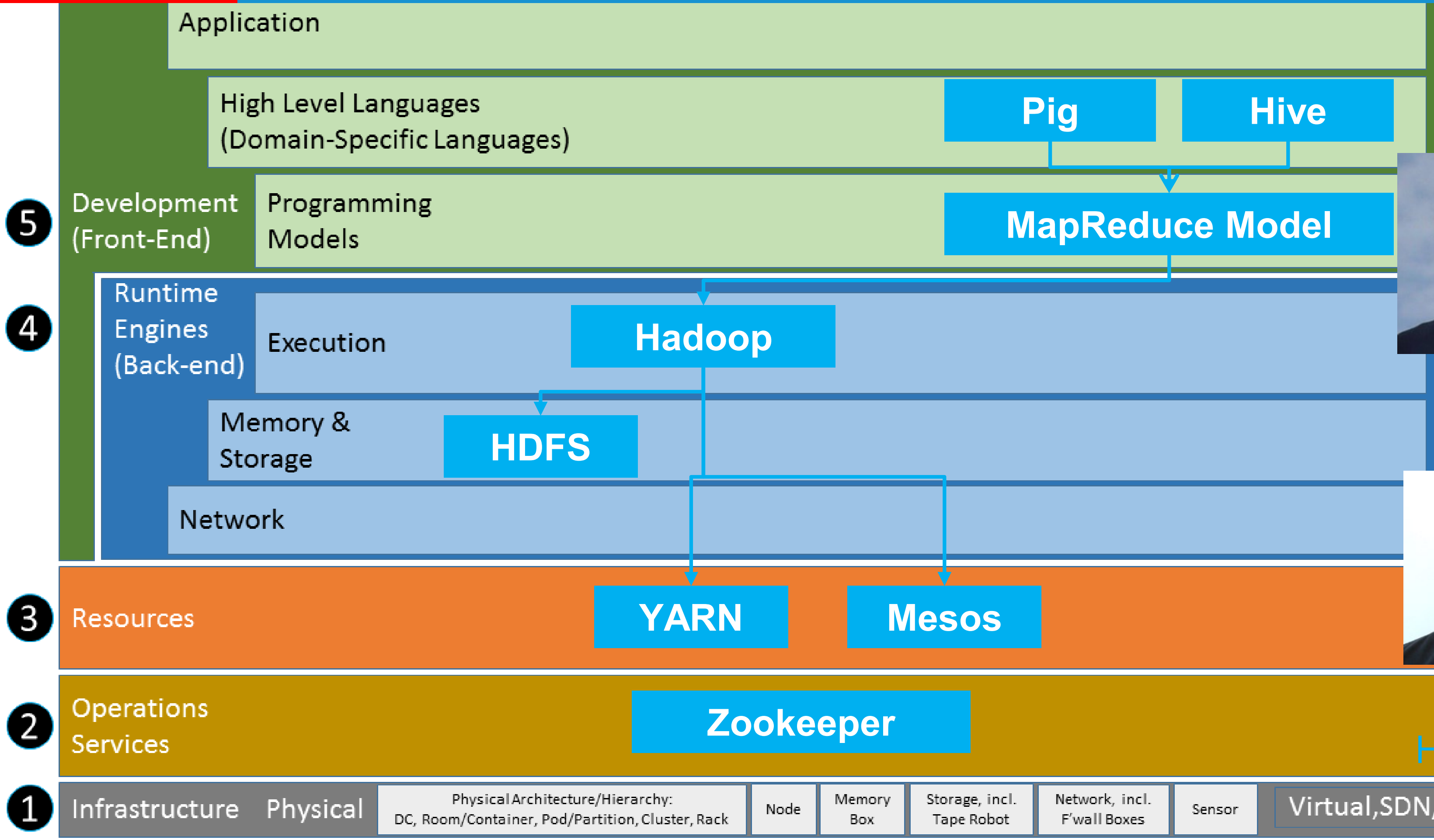
MEANINGFUL DISCOVERY

THE COMPLEXITY CHALLENGE

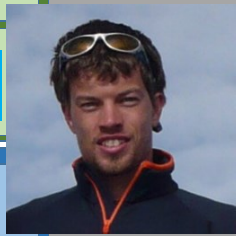
IOSUP ET AL. REFERENCE ARCHITECTURE FOR DCS

Focus on Applications, 5 Core Layers:

5. Development (Front-end)
4. Runtime Engines (Back-end)
3. Resources
2. Operations Services
1. Infrastructure



Tools
100% Dev
2-5'



Vincent van Beek

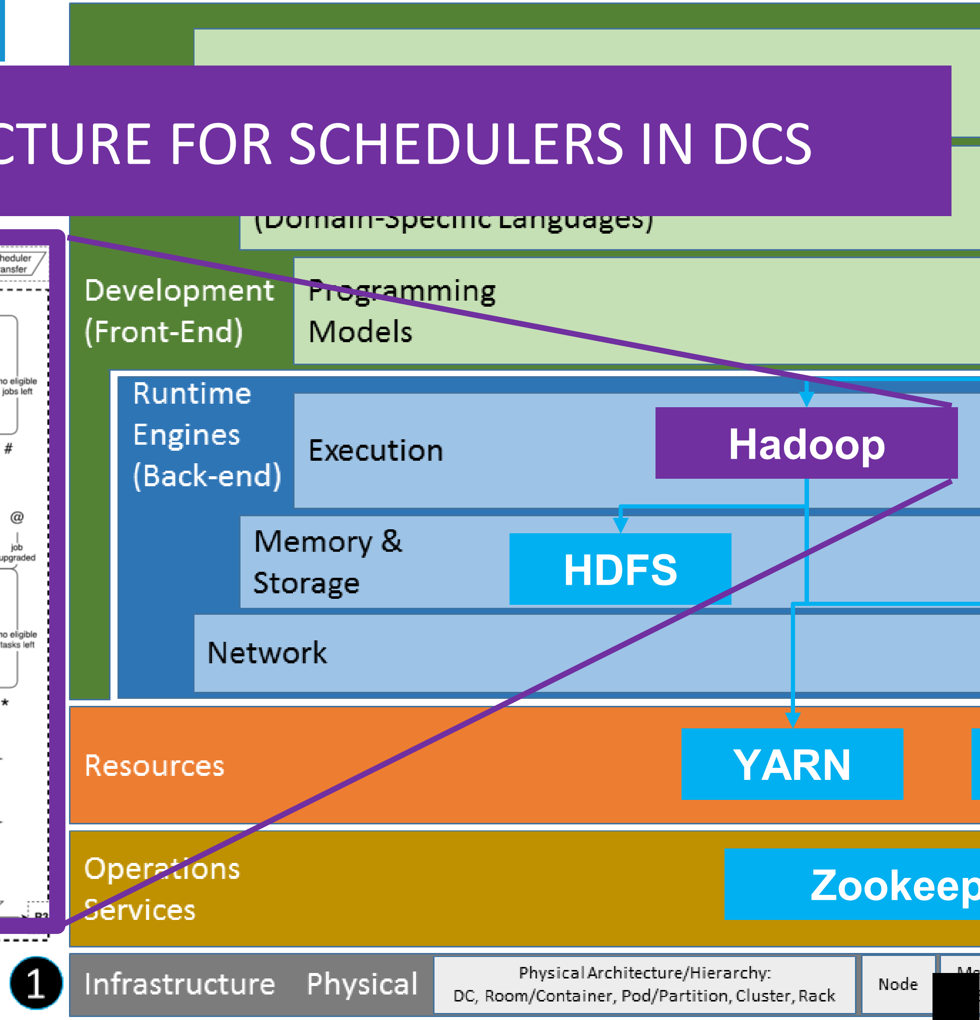
Tim Hegeman

[Iosup et al. ICDCS'18]

MEANINGFUL DISCOVERY

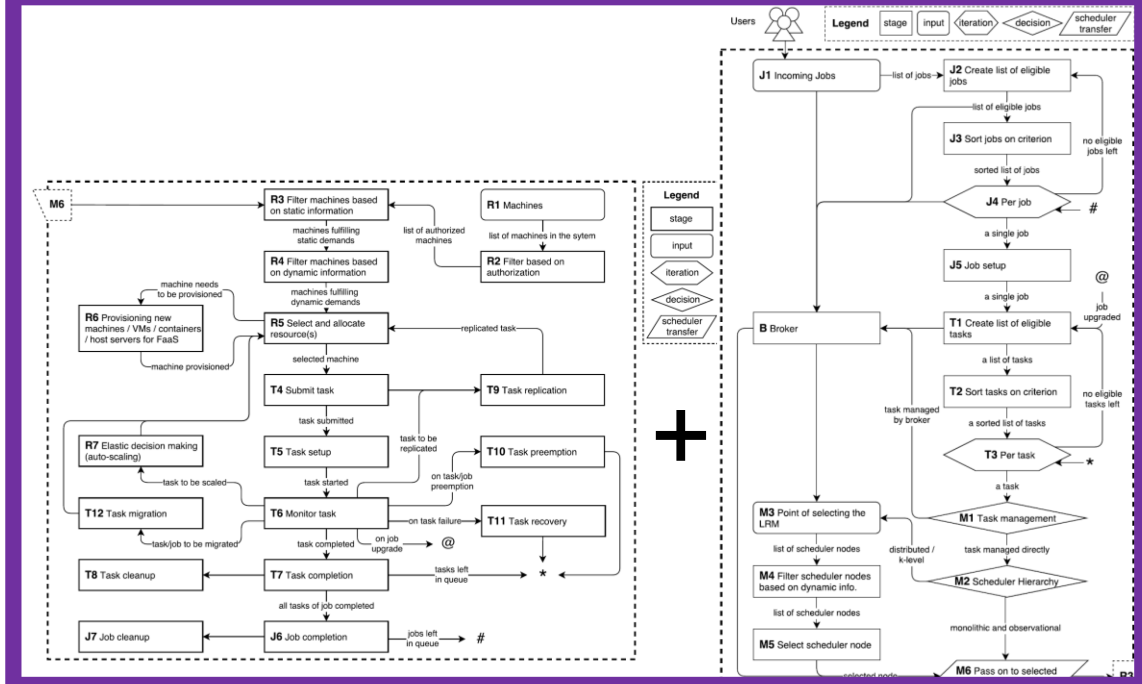
THE COMPLEXITY CHALLENGE

IOSUP ET AL. REFERENCE ARCHITECTURE FOR DCS



ANDREADIS ET AL. REFERENCE ARCHITECTURE FOR SCHEDULERS IN DCS

Georgios Andreadis



[Andreadis et al. SC'18]

Agenda

1. Why this course?
2. A Theory of Distributed Systems
3. A History of Distributed Systems
4. The Science of Distributed Systems
5. **The Virtuous Cycle: A Science-Design-Engineering Example**
6. Take-Home Message

MEANINGFUL DISCOVERY

science + engineering + design

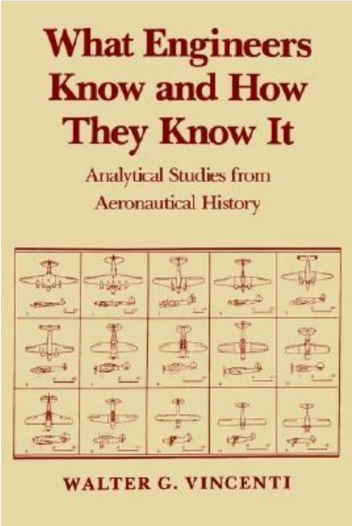
[Iosup et al. ICDCS'18]

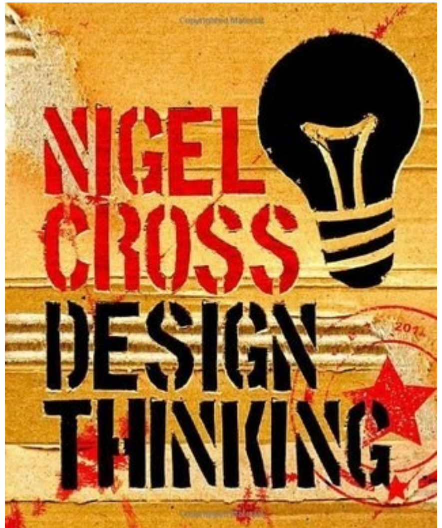
MEANINGFUL DISCOVERY

NO SYSTEMATIC PROCESS FOR COMPUTER SYSTEMS

SO I'LL USE EXAMPLES

science + engineering + design



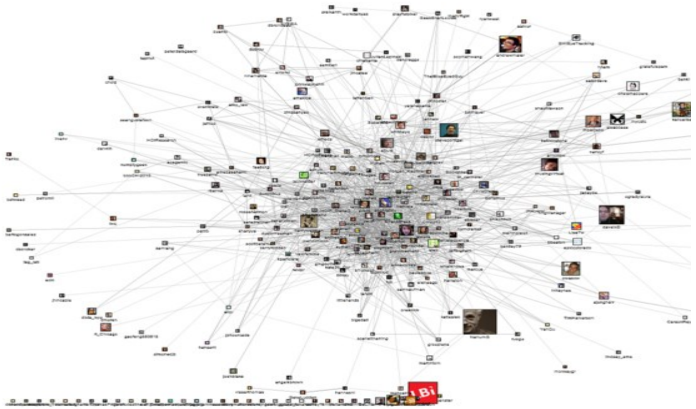


THE COMPUTER SYSTEMS TRIPLET

[Iosup et al. ICDCS'18]

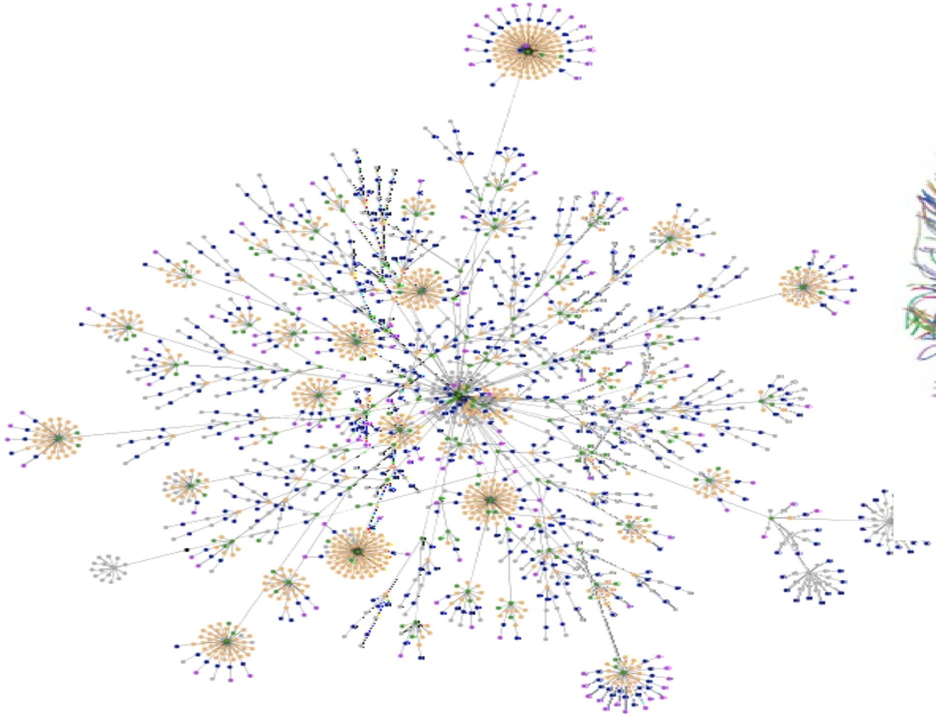
MEANINGFUL DISCOVERY

ENGINEERING LDBC GRAPHALYTICS: THE NEED FOR SPEED … & GRAPHS!



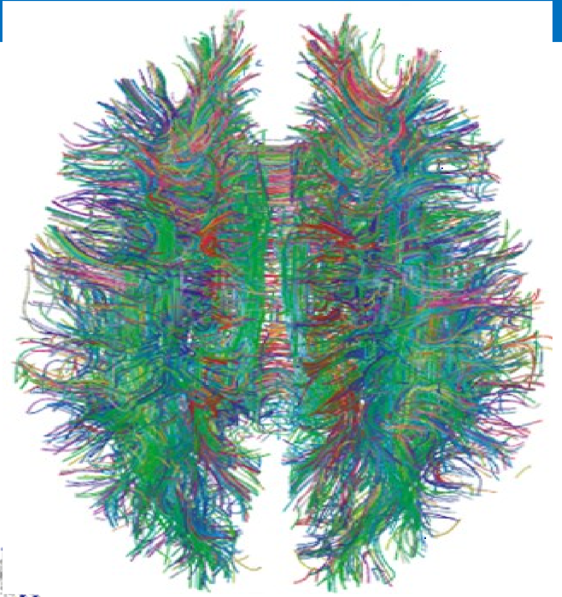
Social network

~1 billion vertices
~100 billion connections



We

~50 billion pages
~1 trillion hyperlinks



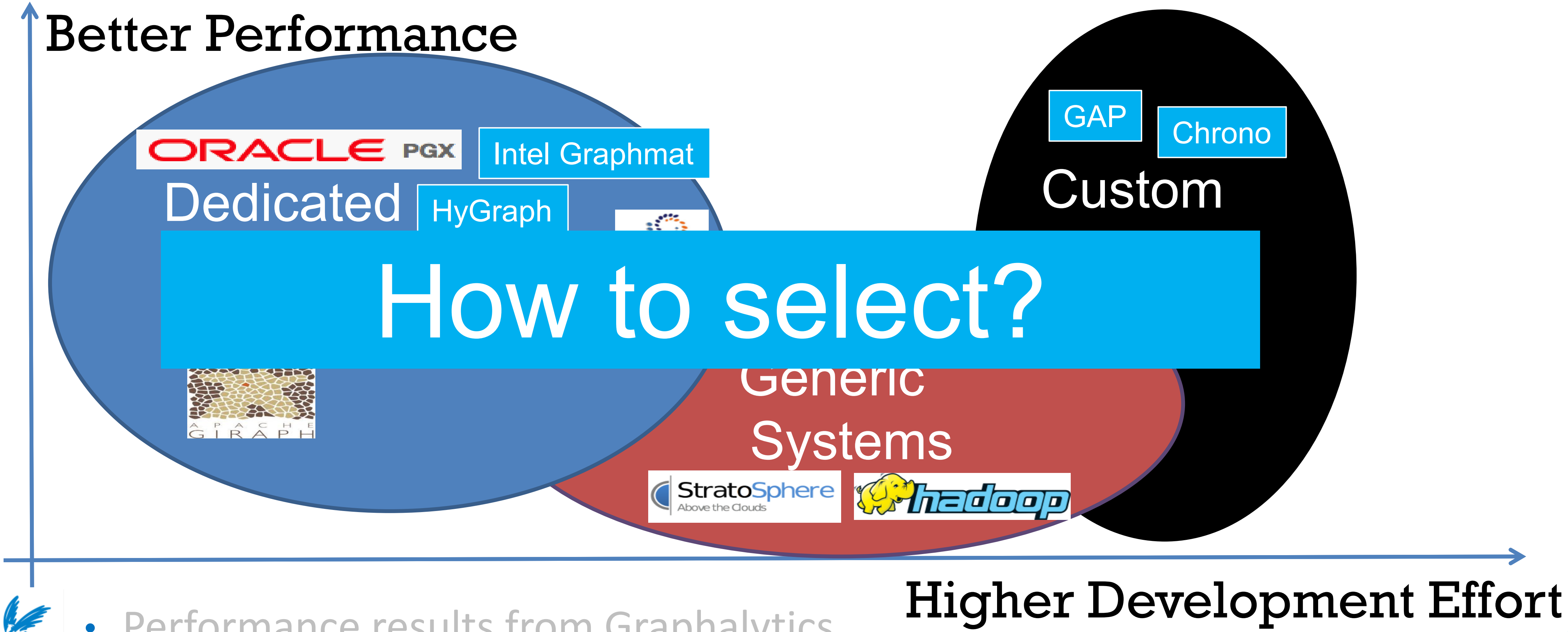
Brain network

~100 billion neurons
~100 trillion connections

Sources: Smith, CHI’10; Blog webpage; Gigandet et al., PLoS ONE 3(12)]

MEANINGFUL DISCOVERY

ENGINEERING LDBC GRAPHALYTICS: THE SYSTEMS LANDSCAPE



- Performance results from Graphalytics

MEANINGFUL DISCOVERY

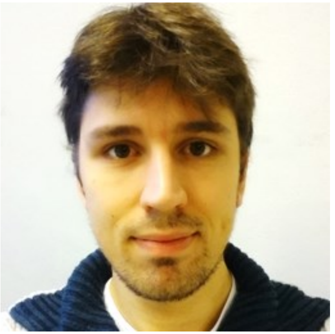
ENGINEERING LDBC GRAPHALYTICS: BENCHMARKING LEADING TO SCIENCE

LDBC *The graph & RDF benchmark reference*

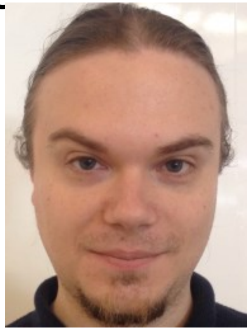
- Graphalytics:
  - > Benchmark
  - > Many classes of algorithms used in practice
  - > Diverse real and synthetic datasets
  - > Diverse experiments, representative for practice
  - > Renewal process to keep the workload relevant
  - > Enables comparison of many platforms, community-driven and industrial
  - > Global Competition

Wing Lung Ngai

Tim Hegeman

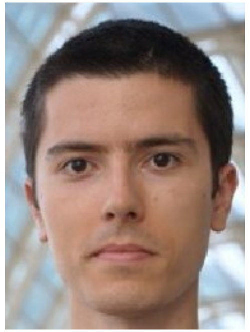


Stijn Heldens



Alex Uță

Ahmed Musaafir



Mihai Capotã

[Iosup et al. PVLDB'16] [Guo et al. CCGRID'15] [Guo et al. IPDPS'14]

MEANINGFUL DISCOVERY

ENGINEERING LDBC GRAPHALYTICS: BENCHMARKING LEADING TO SCIENCE

LDBC *The graph & RDF benchmark reference*

- Graphalytics:
  - > Benchmark
  - > Many classes of algorithms used in practice
  - > Diverse real and synthetic datasets
  - > Diverse experiments, representative for practice
  - > Renewal process to keep the workload relevant
  - > Enables comparison of many platforms, community-driven and industrial
  - > Global Competition

- Community endorsed:

graphalytics.org

- Surprising findings:

Performance: orders of magnitude difference due to each of platform, algorithm, dataset, and hardware

- Triggered new research

[Iosup et al. PVLDB'16] [Guo et al. CCGRID'15] [Guo et al. IPDPS'14]

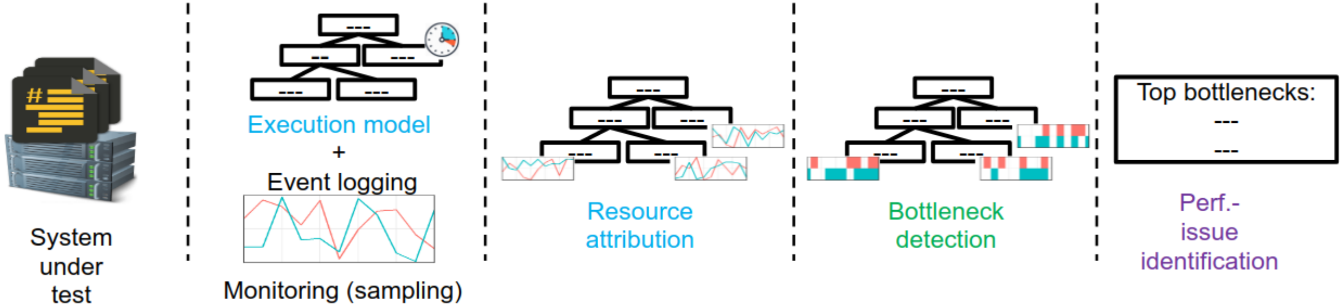
MEANINGFUL DISCOVERY

ENGINEERING LDBC GRAPHALYTICS: MODELING LEADS TO PERFORMANCE ANALYSIS

LDBC *The graph & RDF benchmark reference*

- Graphalytics Grade10:
  > Automated bottleneck detection
  > Automated identification of performance issues

Tim Hegeman



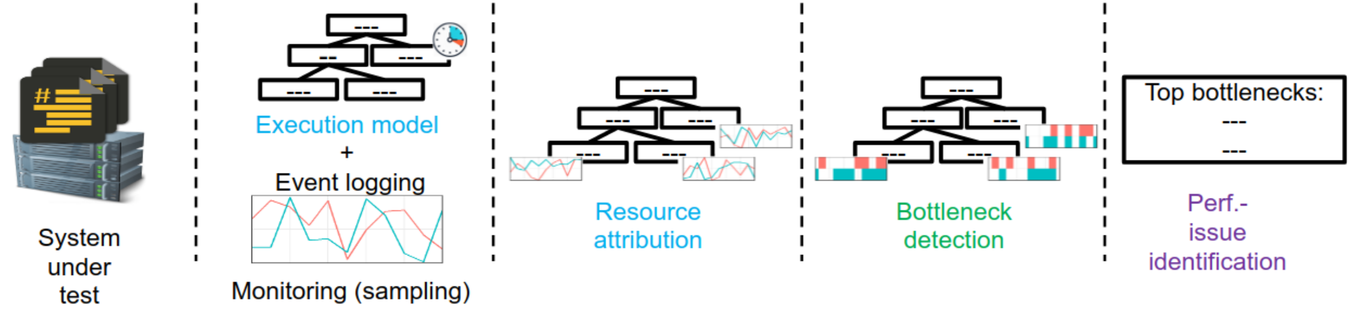
Multi-stage process, works in ecosystem

MEANINGFUL DISCOVERY

ENGINEERING LDBC GRAPHALYTICS: MODELING LEADS TO PERFORMANCE ANALYSIS

LDBC *The graph & RDF benchmark reference*

- Graphalytics Grade10:
  - > Automated bottleneck detection
  - > Automated identification of performance issues



Multi-stage process, works in ecosystem

APACHE GIRAPH

- Without Grade10:

No bottleneck at all

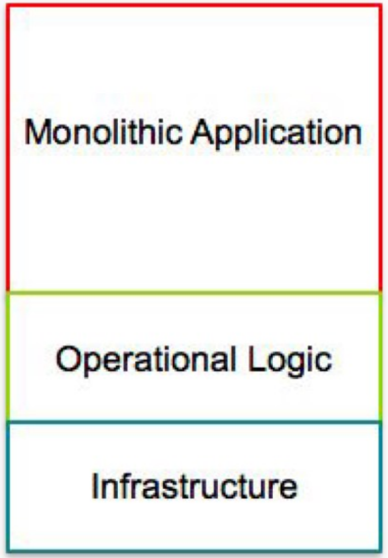
- With Grade10:

Always bottleneck
Cause:
+ Message queue full
+ Garbage collector
+ CPU
+ Others

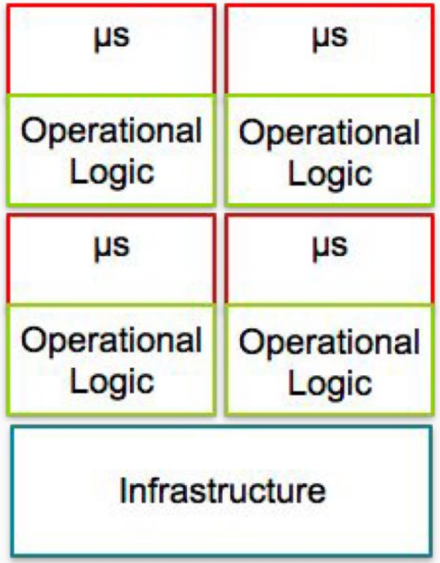
MEANINGFUL DISCOVERY

DESIGNING SERVERLESS ARCHITECTURES

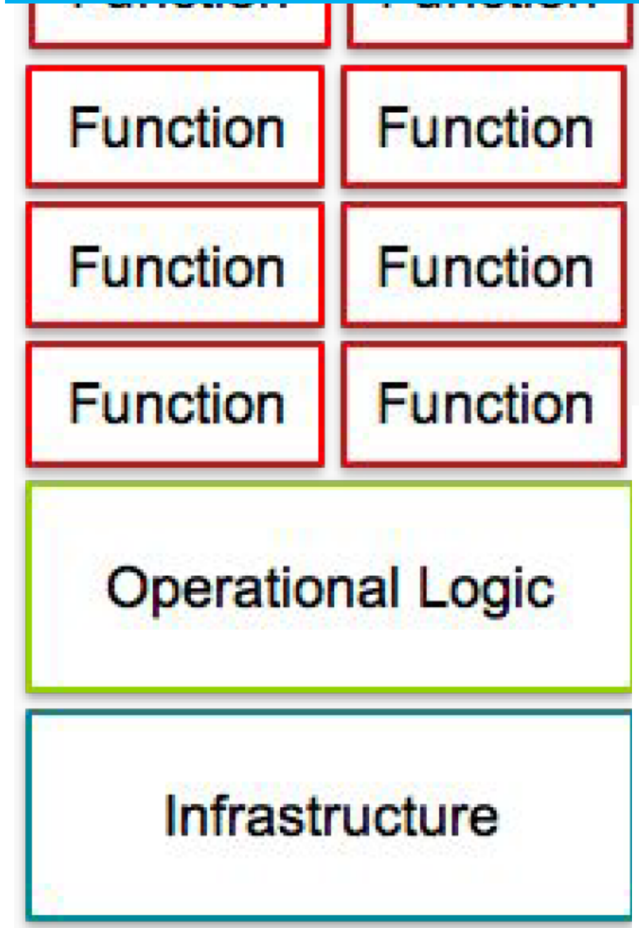
Abstraction: Serverless
Design: FaaS systems

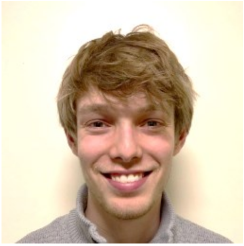


Difficult to Scale
Infrequent, Inflexible
Complex deployment
Tightly coupled stack



- Scalable
- Frequent, Flexible
- Complexity: from application logic to operational logic





Erwin van Eyk

Lucian Toader

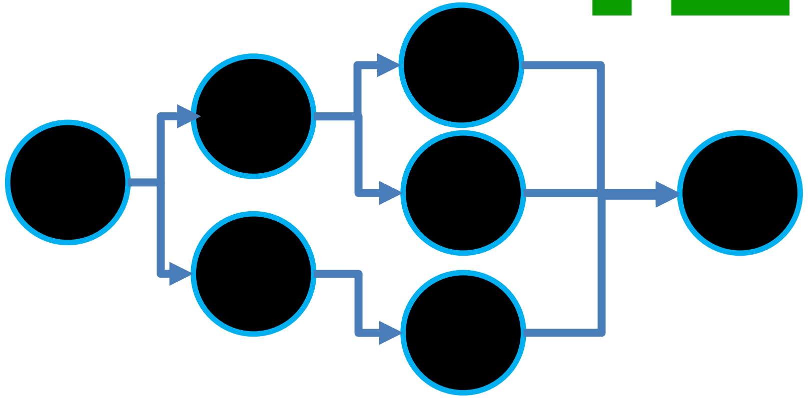
- Scalable
- Frequent, Flexible
- Explicit separation of Business Logic vs. Operational Logic.
- Minimal layer coupling, unit of deployment

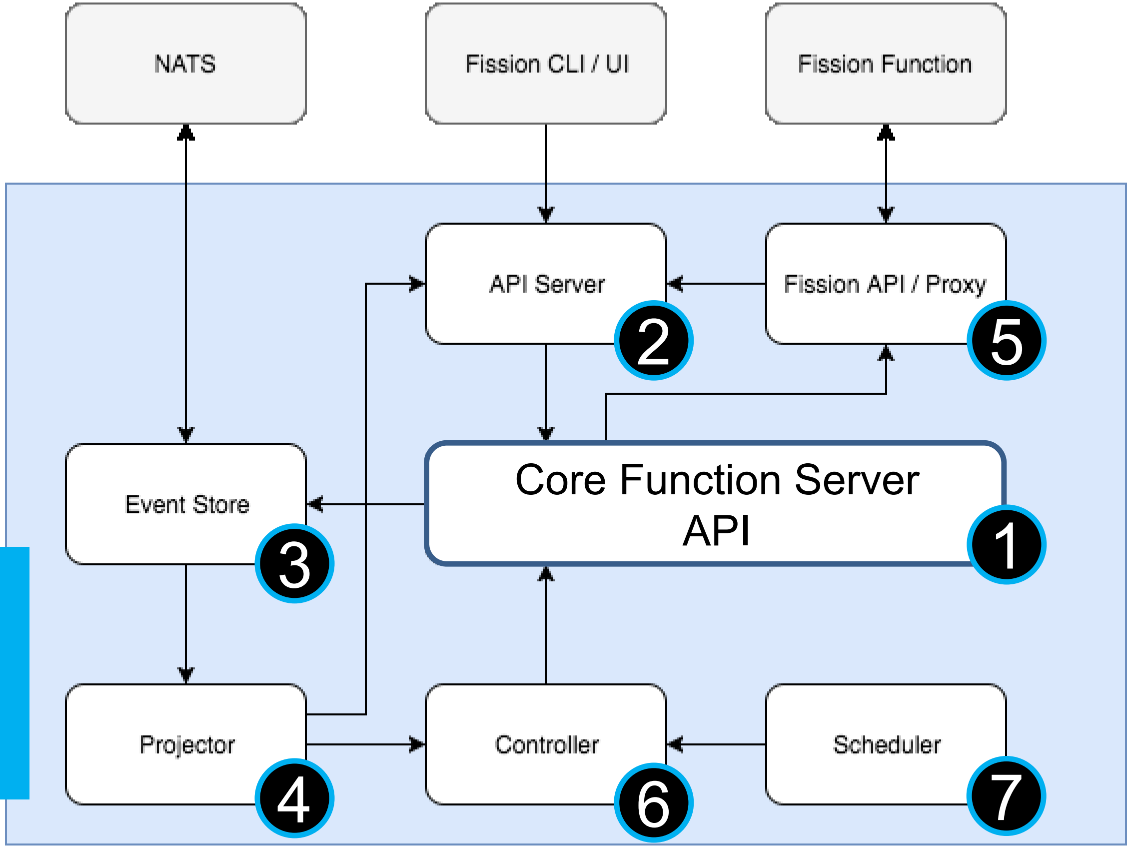
VU

MEANINGFUL DISCOVERY

DESIGNING SERVERLESS ARCHITECTURES

Erwin
van Eyk





The workflow management engine,
part of the Serverless ecosystem
at Fission.io

https://github.com/fission/fission-workflows/blob/master/Docs/architecture.md

Agenda

1. Why this course?
2. A Theory of Distributed Systems
3. A History of Distributed Systems
4. The Science of Distributed Systems
5. The Virtuous Cycle: a Science-Design-Engineering Example
6. **Take-Home Message**

~~Conclusion~~ Take-Home Message

- Golden Age of distributed systems and ecosystems

- A theory of distributed systems and ecosystems
- The science of distributed ecosystems
- Science-Design-Engineering virtuous cycle
- (Ask about the challenges)

http://www.flickr.com/photos/dimitrisotiropoulos/4204766418

- Reality Check: we are all clients, from e-Science, to GAFAM and BAT, from Twitter and Netflix, to banking and MMOs

FURTHER READING

1. Iosup et al. Massivizing Computer Systems. ICDCS 2018 ← start here
2. Andreadis et al. A Reference Architecture for Datacenter Scheduling, SC18
3. Van Eyk et al. Serverless is More: From PaaS to Present Cloud Computing, IEEE IC Sep/Oct 2018 (in print)
4. Uta et al. Exploring HPC and Big Data Convergence: A Graph Processing Study on Intel Knights Landing, IEEE Cluster 2018

5. Jiang et al. Mirror. CCPE 2018.
6. Ilyushkin et al. Autoscaling for Complex Workflows. TOMPECS 2018.
7. Iosup et al. The OpenDC Vision. ISPDC'17.
8. Iosup et al. Self-Aware Computing Systems book, 2017.
9. Iosup et al. LDBC Graphalytics. PVLDB 2016.
10. Guo et al.: Heterogeneous Graph-Processing. CCGrid 2016.

11. van Beek et al.: IEEE Computer 2015.
12. Jia et al.: TKDD 2015.
13. Ghit et al. SIGMETRICS 2014.
14. Iosup and Epema: IEEE Internet Computing 2011.
15. Iosup et al.: CCGRID 2011.
16. Iosup et al.: IEEE TPDS 2011.

FURTHER READING II

17. Javadi, Kondo, Iosup, Epema (2013) The Failure Trace Archive: Enabling the comparison of failure measurements and models of distributed systems. J. Parallel Distrib. Comput. 73(8): 1208-1223.
18. Guo and Iosup: The Game Trace Archive. NetGames 2012: 1-6.
19. Iosup et al. (2008) The Grid Workloads Archive. Future Generation Comp. Syst. 24(7): 672-686.
20. Adele Lu Jia et al. (2016) When Game Becomes Life: The Creators and Spectators of Online Game Replays and Live Streaming. TOMCCAP 12(4): 47:1-24.
21. Shen, van Beek, and Iosup: Statistical Characterization of Business-Critical Workloads Hosted in Cloud Datacenters. CCGRID 2015: 465-474.
22. Adele Lu Jia et al. (2015) Socializing by Gaming: Revealing Social Relationships in Multiplayer Online Games. TKDD 10(2): 11:1-29.
23. Iosup et al. (2014) : Analyzing Implicit Social Networks in Multiplayer Online Games. IEEE Internet Computing 18(3): 36-44 (2014).
24. Zhang et al.: Identifying, analyzing, and modeling flashcrowds in BitTorrent. Peer-to-Peer Computing 2011: 240-249.
25. Yigitbasi et al.: Analysis and modeling of time-correlated failures in large-scale distributed systems. GRID 2010: 65-72.
26. Gallet et al.: A Model for Space-Correlated Failures in Large-Scale Distributed Systems. Euro-Par (1) 2010: 88-100.
27. Iosup, Sonmez, and Epema: DGSim: Comparing Grid Resource Management Architectures through Trace-Based Simulation. Euro-Par 2008: 13-25
28. Guo, Hong, Chafi, Iosup, and Epema (2017) Modeling, analysis, and experimental comparison of streaming graph-partitioning policies. J. Parallel Distrib. Comput. 108: 106-21.
29. Guo et al.: Benchmarking graph-processing platforms: a vision. ICPE 2014: 289-292.
30. Uta et al. Elasticity in Graph Analytics? A Benchmarking Framework for Elastic Graph Processing. IEEE Cluster 2018.
31. Heldens, Varbanescu, Iosup: Dynamic Load Balancing for High-Performance Graph Processing on Hybrid CPU-GPU Platforms. IA3@SC 2016: 62-65.
32. Guo et al.: An Empirical Performance Evaluation of GPU-Enabled Graph-Processing Systems. CCGRID 2015: 423-432.
33. Herbst et al. (2016) Ready for Rain? A View from SPEC Research on the Future of Cloud Metrics. CoRR abs/1604.03470 (2016). (in print in TOMPECS)
34. Deng, Song, Ren, and Iosup: Exploring portfolio scheduling for long-term execution of scientific workloads in IaaS clouds. SC 2013: 55:1-55:12.
35. Shen, Deng, Iosup, and Epema: Scheduling Jobs in the Cloud Using On-Demand and Reserved Instances. Euro-Par 2013: 242-254.