THE DATACENTER NEEDS AN OPERATING SYSTEM

MATEI ZAHARIA, BENJAMIN HINDMAN, ANDY KONWINSKI, ALI GHODSI, ANTHONY JOSEPH, RANDY KATZ, SCOTT SHENKER, ION STOICA

UC BERKELEY

THE DATACENTER IS THE NEW COMPUTER

Running today's most popular consumer apps

- Facebook, Google, iCloud, etc

Needed for big data in business & science

Widely accessible through cloud computing

Our claim: **this new computer needs an operating system**

WHY DATACENTERS NEED AN OS

Growing diversity of applications

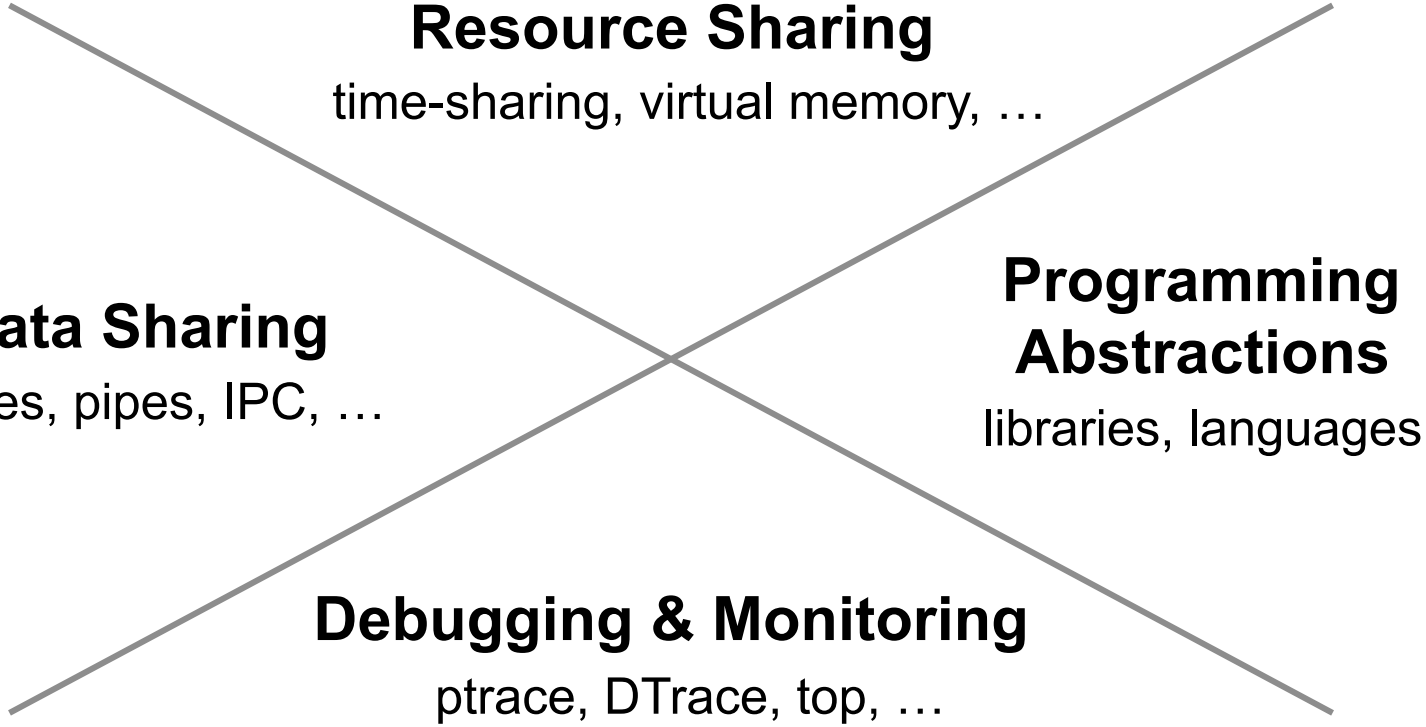
- Computing frameworks: MapReduce, Dryad, Pregel, Percolator, Dremel
- Storage systems: GFS, BigTable, Dynamo, etc

Growing diversity of users

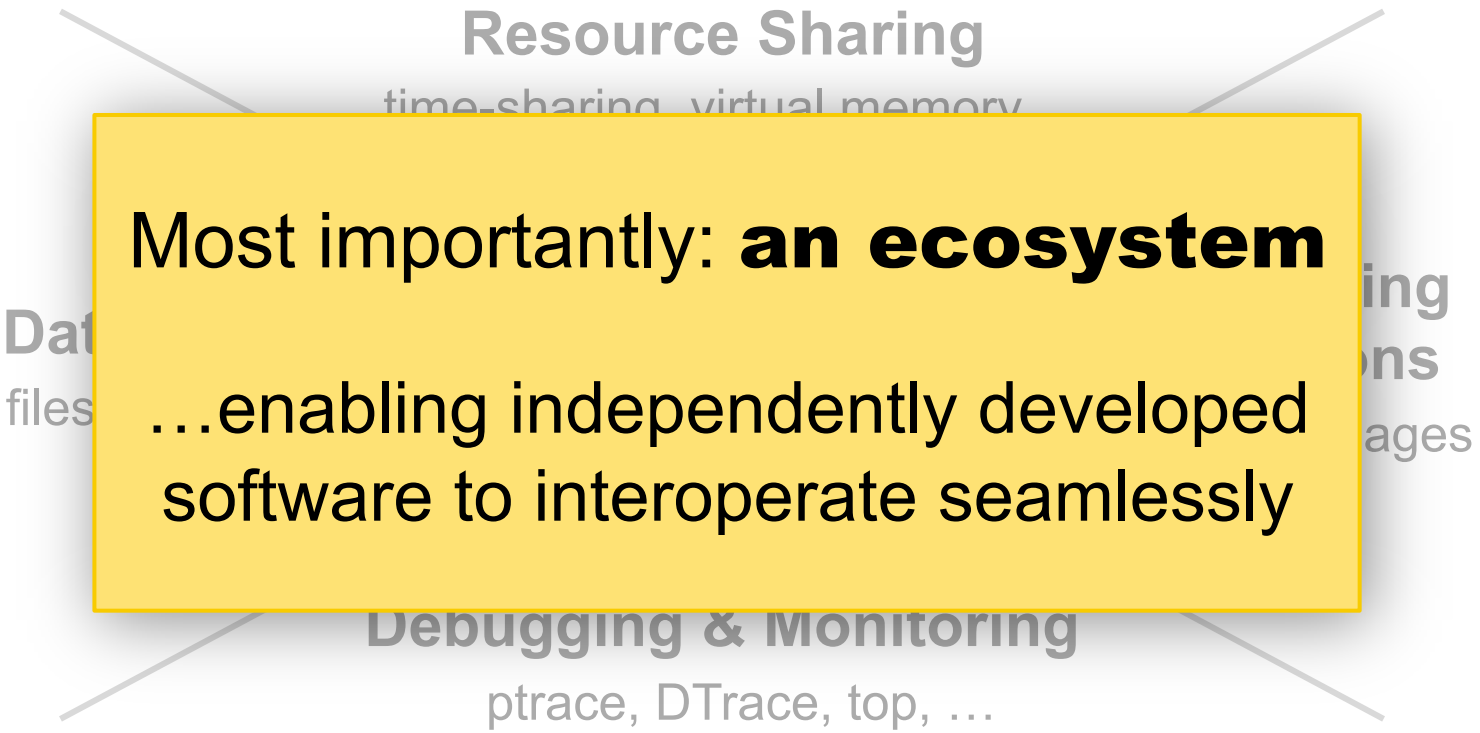
- 200+ Hive users at Facebook

Same reasons computers needed one!

WHAT OPERATING SYSTEMS PROVIDE



WHAT OPERATING SYSTEMS PROVIDE



Most importantly: **an ecosystem**

…enabling independently developed software to interoperate seamlessly

TODAY'S DATACENTER OPERATING SYSTEM

Platforms like Hadoop well-aware of these issues

- Inter-user resource sharing, but at the level of MapReduce jobs (though this is changing)
- InputFormat API for storage systems (but what happens with the next hot platform after Hadoop?)

Other examples: Amazon services, Google stack

TODAY'S DATACENTER OPERATING SYSTEM

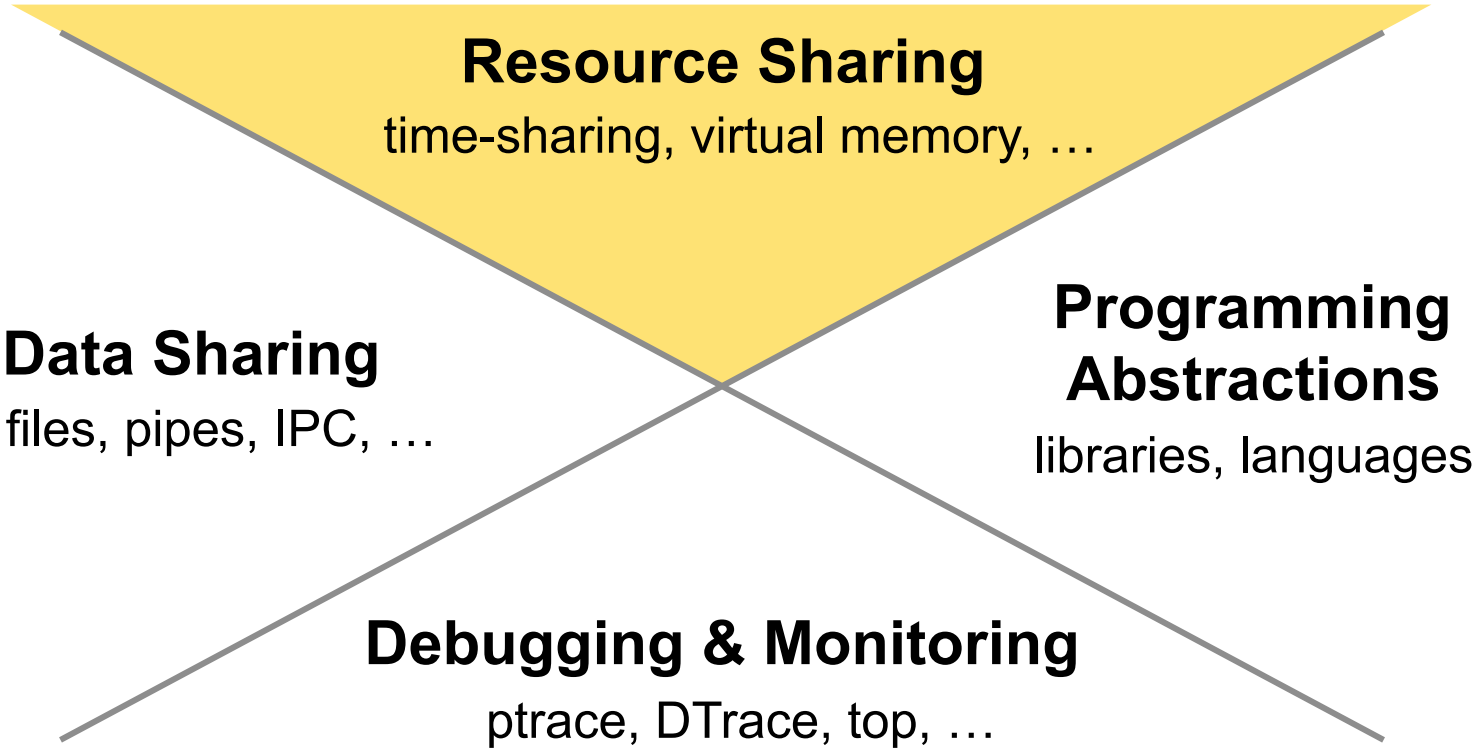
Platforms like Hadoop well-aware of these issues

- Inter-user resource sharing, but at the level of

The **problems** motivating a datacenter OS are well recognized, but solutions are **narrowly targeted**

Can researchers take a longer-term view?

TOMORROW'S DATACENTER OS



RESOURCE SHARING

"To solve these interaction problems we would like to have a computer made simultaneously available to many users in a manner somewhat like a telephone exchange. Each user would be able to use a console at his own pace and without concern for the activity of others using the system."

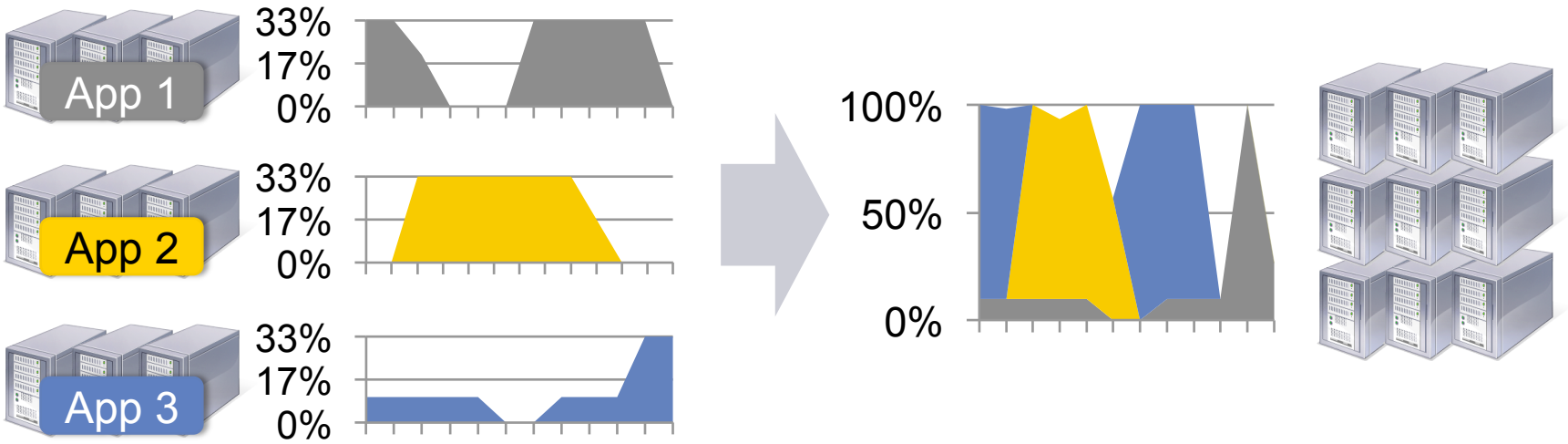
– Fernando J. Corbató, 1962

RESOURCE SHARING

Today, cluster apps are built to run independently and assume they own a fixed set of nodes

Result: inefficient static partitioning

What's the right interface for dynamic sharing?



MEMORY MANAGEMENT

Memory is an increasingly important resource

- In-memory iterative processing (Pregel, Spark, etc)
- DFS cache for MapReduce cluster could serve 90% of jobs at Facebook (HotOS ‘11)

What are the right memory management algorithms for a parallel analytics cluster?

PROGRAMMING AND DEBUGGING

Although there are new programming models for applications, system programming remains hard

- Can we identify useful common abstractions? (Chubby, Sinfonia, Mesos are some examples)
- How much can languages (e.g. Go, Erlang) help?

Debugging is *very* hard

- Magpie, X-Trace, Dapper are some steps here

Can a clean-slate design of the stack help?

HOW RESEARCHERS CAN HELP

Focus on paradigms, not only performance

- Industry is spending a lot of time on performance

Explore clean-slate approaches

- Much datacenter software is written from scratch
- People using Erlang, Scala, functional models (MR)

Bring cluster computing to non-experts

- Most impactful (datacenter as the new workstation)
- Hard to make a Google-scale stack usable without a Google-scale ops team

CONCLUSION

Datacenters are becoming a major platform

To support a thriving software ecosystem like computers do, they need the equivalent of an OS

Researchers can take a **long-term** systems view to problems arising today to enable this