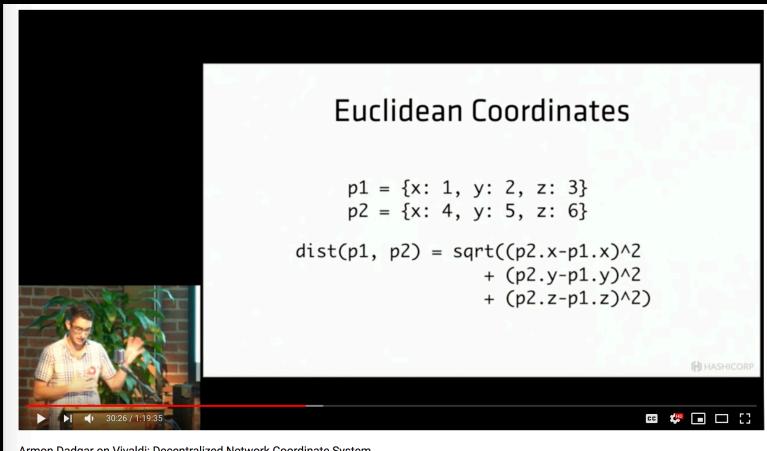




Research as Code

We Talk A Lot About Our Use of Research

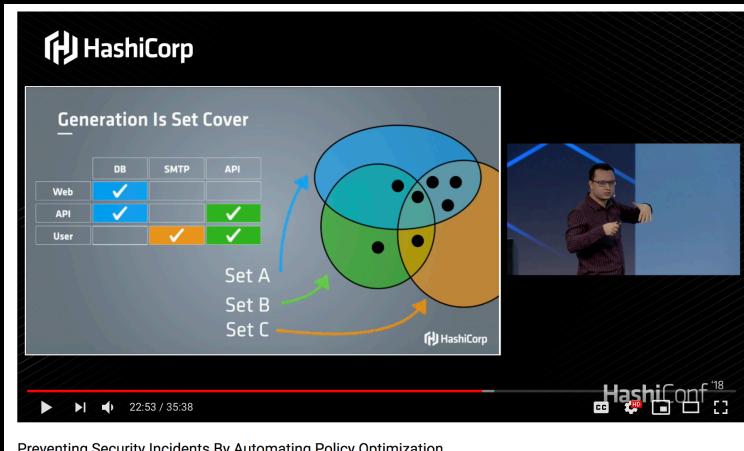


Armon Dadgar on Vivaldi: Decentralized Network Coordinate System

Euclidean Coordinates

```
p1 = {x: 1, y: 2, z: 3}
p2 = {x: 4, y: 5, z: 6}

dist(p1, p2) = sqrt((p2.x-p1.x)^2
+ (p2.y-p1.y)^2
+ (p2.z-p1.z)^2)
```



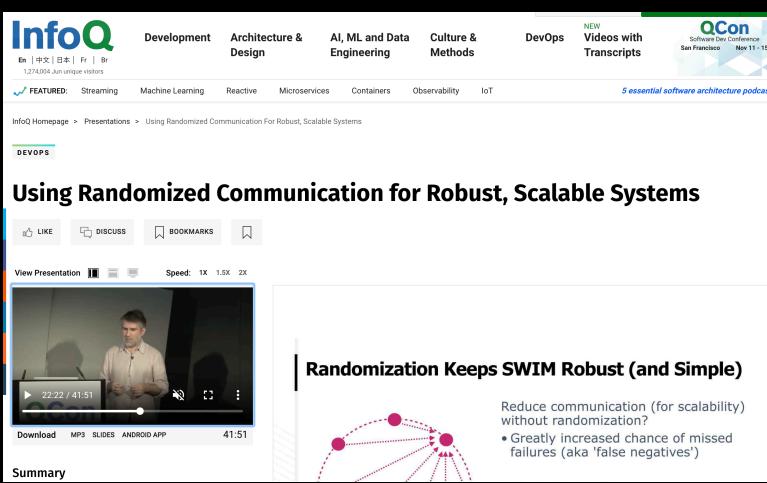
HashiCorp

Generation Is Set Cover

	DB	SMTP	API
Web	✓		
API	✓		✓
User		✓	✓

Set A
Set B
Set C

Preventing Security Incidents By Automating Policy Optimization



InfoQ

Development Architecture & Design AI, ML and Data Engineering Culture & Methods DevOps NEW Videos with Transcripts

FEATURED Streaming Machine Learning Reactive Microservices Containers Observability IoT

5 essential software architecture podcasts

InfoQ Homepage > Presentations > Using Randomized Communication For Robust, Scalable Systems

DEVOPS

Using Randomized Communication for Robust, Scalable Systems

LIKE DISCUSS BOOKMARKS

View Presentation Speed: 1X 1.5X 2X

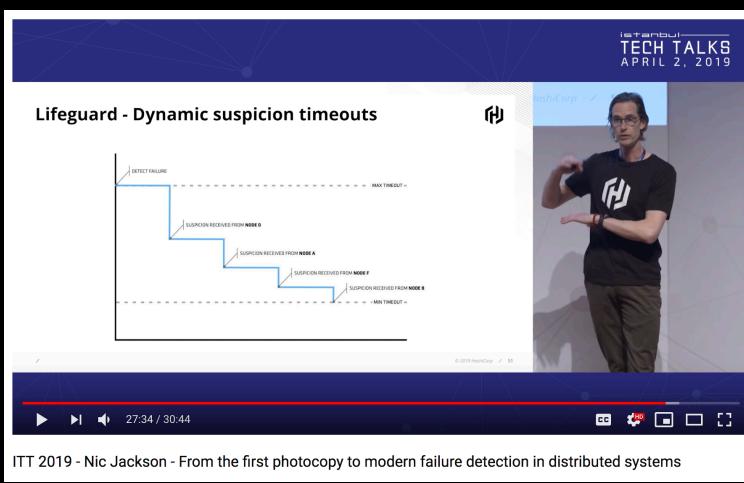
22.22 / 41.51

Download MP3 SLIDES ANDROID APP 41.51

Summary

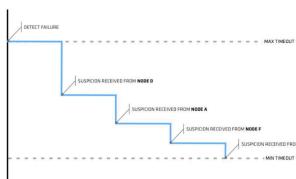
Randomization Keeps SWIM Robust (and Simple)

Reduce communication (for scalability) without randomization?
• Greatly increased chance of missed failures (aka 'false negatives')



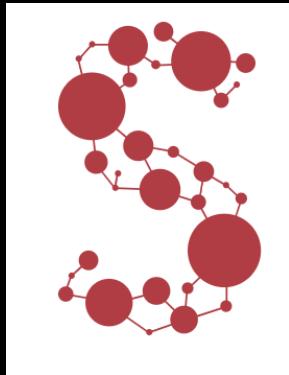
TECH TALKS APRIL 2, 2019

Lifeguard - Dynamic suspicion timeouts



ITT 2019 - Nic Jackson - From the first photocopy to modern failure detection in distributed systems

Engaged with Research Since Day 1



Serf

A screenshot of a GitHub repository page. The repository name is "hashicorp / memberlist". The top navigation bar shows "Code" (selected), Issues (37), Pull requests (10), Actions, Projects (0), Wiki, Security, Insights, and Settings. Below the navigation bar, it says "Branch: master". A collapsible section titled "Commits on Sep 9, 2013" is expanded, showing two commits: "Adding a gitignore file" by armon committed on Sep 9, 2013, with commit hash 9928b17; and "Initial commit" by armon committed on Sep 9, 2013, with commit hash 1a09a04.

hashicorp / memberlist

Code Issues 37 Pull requests 10 Actions Projects 0 Wiki Security Insights Settings

Branch: master

Commits on Sep 9, 2013

Adding a gitignore file
armon committed on Sep 9, 2013

Initial commit
armon committed on Sep 9, 2013

Chose to implement **SWIM** ...

But only after careful evaluation of **Plumtree**, **T-Man** and **HyParView**

Actually, Since Way Before Day 1



← → ⌂ <https://www.cs.washington.edu>

**W PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING**

Retaining Sandbox Containment Despite Bugs in Privileged Memory-Safe Code

Justin Cappos, Armon Dadgar, Jeff Rasley, Justin Samuel, Ivan Beschastnikh, Cosmin Barsan, Arvind Krishnamurthy, Thomas Anderson

Department of Computer Science and Engineering
University of Washington
Seattle, WA 98195

{justinc,armond,jeffra45,jsamuel,ivan,cosminb,arvind,tom}@cs.washington.edu

Abstract

Flaws in the standard libraries of secure sandboxes represent a major security threat to billions of devices worldwide. The standard libraries are hard to secure because they frequently need to perform low-level operations that are forbidden in untrusted application code. Existing designs have a single, large trusted computing base that contains security checks at the boundaries between trusted and untrusted code. Un-

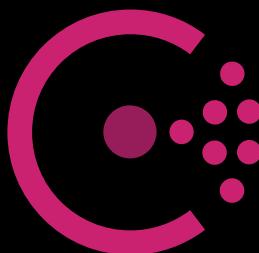
1. INTRODUCTION

Programming language sandboxes, such as Java, Silverlight, JavaScript, and Flash, are ubiquitous. Such sandboxes have gained widespread adoption with web browsers, within which they are used for untrusted code execution, to safely host plug-ins, and to control application behavior on closed platforms such as mobile phones. Despite the fact that program containment is their primary goal, flaws in

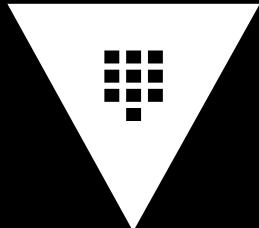


Research

Fair to Say: HashiCorp ❤ Research!



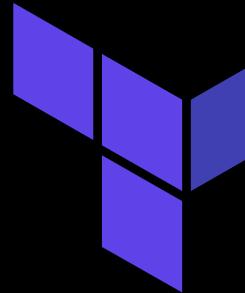
- Consensus (Raft)
- Gossip (SWIM, Lifeguard)
- Lamport and Vector Clocks
- Multi-Version Concurrency Control (MVCC)
- Network Coordinates (Vivaldi & follow ups)



- Secure Authentication (Kerberos)
- Security Protocols
- Capability-Based Security
- Access Control Management
- Cryptography



- Scheduler Design (Mesos, Borg, Omega)
- Bin Packing
- Pre-emption
- Randomized choice



- Graph Theory
- Type Theory
- Automata Theory

Heavy Research Use Leads To Doing Research! ↗

Combining multiple pieces of research ...

Supporting them at scale unforeseen by the authors ...

Leads to adapting and extending the designs

Just add principled evaluation and prior work



Ready to 'give back', based on the experience gained adapting existing research

An industrial research lab, with dual mandate:

- Have impact on HashiCorp's open source tools and enterprise products**
- Contribute to the academic and practitioner community with papers and code**

Academic (aka 'Proper') Research



Standard is peer-reviewed publication

Novel contribution to the literature

Must consider *all* relevant prior work

Adhere to The Scientific Method

Consume More, Not Less Existing Research

Any novel solution we offer for publication must be compared to all relevant alternatives, and justified in terms of deficiencies they have in real-world situations

This is the only way the community can make coherent progress, so it is enforced strictly in review



Still Use Lots of Existing Research

Adopting existing solutions should still be our first choice, so long as they satisfy our users' real-world needs

"Not invented here" will just slow us down, and keep us from working on solving novel problems



Today's Topic: Research Methods ...

How do we reliably discover all relevant research for a project?

How do we make sure we evaluate the research reliably against our users' actual needs?

... In Support Of Product Development

How do we keep all stakeholders (engineering, support, sales, research, ...) on the same page regarding requirements?

How do we empower engineering teams to evaluate research themselves, as far as possible?



Research as Code

High-Level Approach



Agile, Artifact-Driven Research Workflows

Use Product Development Tools and Processes

Research Should Adopt Agile,

- Collaborative research discovery culture**
- Agile research projects**

-



github.com/hashicorp/research-resources

404 until end of the talk :-)



The HashiCorp Approach

-
- Belief in The Scientific Method
-
- Collaborative research discovery culture
- Agile research projects
-



Thank You



**Body slide
3 line max.**



**Body slide
3 line max.**

Jon Currey - Industrial Researcher



≡ Google Scholar



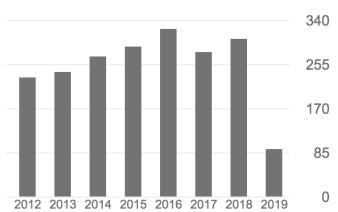
Jon Currey
HashiCorp Research
Verified email at hashicorp.com
[Distributed Systems and M...](#)

[FOLLOW](#) [GET MY OWN PROFILE](#)

TITLE	CITED BY	YEAR
DryadLINQ: A system for general-purpose distributed data-parallel computing using a high-level language YYMID Fetterly, M Budiu, Ú Erlingsson, PKGJ Currey Proc. LSDS-IR 8	902	2009
Quincy: fair scheduling for distributed computing clusters M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, A Goldberg Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles ...	856	2009
An introduction to computational networks and the computational network toolkit D Yu, A Eversole, M Seltzer, K Yao, Z Huang, B Guenter, O Kuchaiev, ... Microsoft Technical Report MSR-TR-2014-112	245	2014
PTask: operating system abstractions to manage GPUs as compute devices CJ Rossbach, J Currey, M Silberstein, B Ray, E Witchel Proceedings of the Twenty-Third ACM Symposium on Operating Systems ...	230	2011
Dandelion: a compiler and runtime for heterogeneous systems CJ Rossbach, Y Yu, J Currey, JP Martin, D Fetterly Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems ...	118	2013
System and method for logging messages in an embedded computer system JJ Currey, JE Rodriguez, RM Jensen, KR Black US Patent 6,769,079	22	2004
Some sample programs written in DryadLINQ Y Yu, M Isard, D Fetterly, M Budiu, Ú Erlingsson, PK Gunda, J Currey, ... Tech. Rep. MSR-TR-2008-74, Microsoft Research	15	2008
System and method for adding transport protocols in distributed middleware applications KR Black, RM Jensen, JE Rodriguez, JJ Currey US Patent 7,010,609	14	2006

Cited by [VIEW ALL](#)

All	Since 2014
Citations 2453	1562
h-index 9	6
i10-index 9	5



Co-authors

 CHRISTOPHER J. ROSSBACH University of Texas at Austin and...
 Dennis Fetterly Software Engineer, Google
 Úlfar Erlingsson Research Scientist, Google Brain
 Michael Isard Research Scientist, Google
 Andrew V. Goldberg Senior Principal Scientist, Amaz...

Citations and References



I. INTRODUCTION

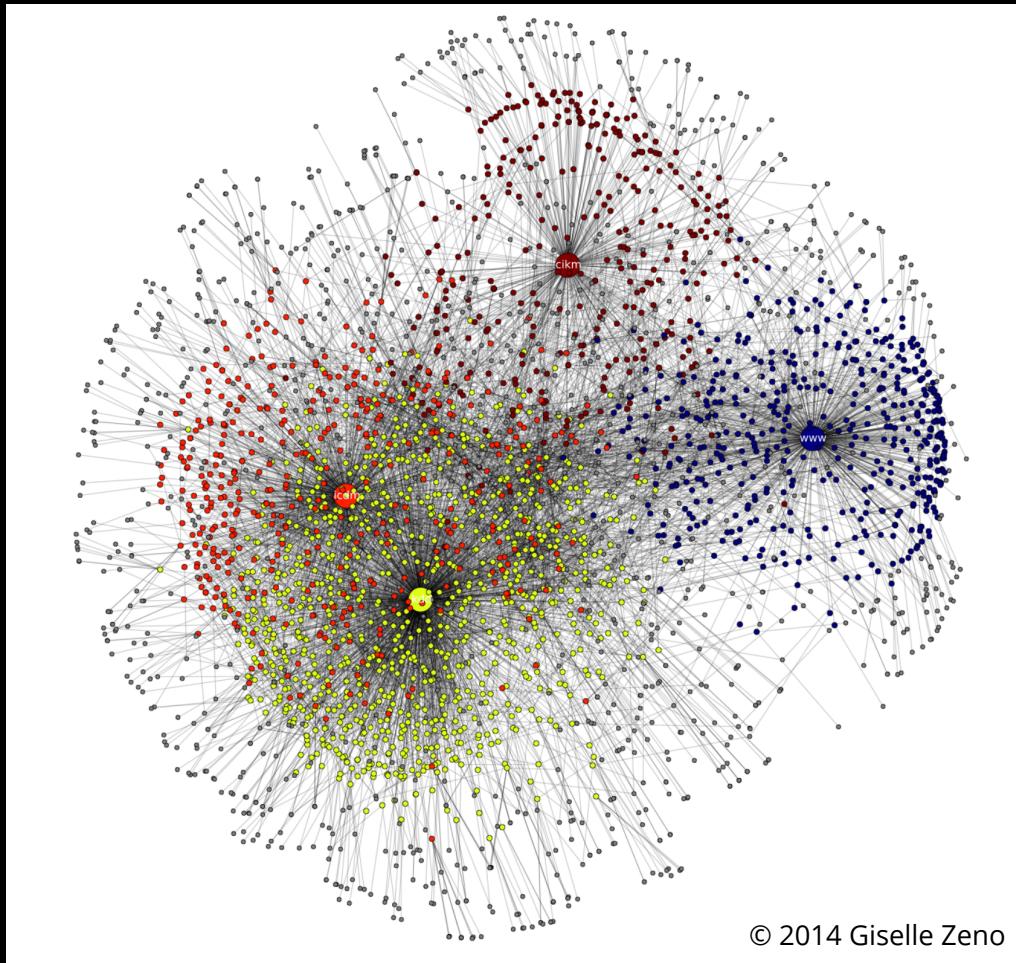
Group membership is an intuitive abstraction that can be used to address discovery, failure detection, and load balancing of components of a distributed system. SWIM [1] is a group membership protocol with a simple peer-to-peer design. Its use of randomized communication make it highly scalable, robust to both node and network failures, and easy to manage.

We are aware of three mature open source implementations of SWIM. Butterfly [2] is part of Habitat [3], a popular software automation platform. Ringpop [4] was built to support the applications of a global transportation technology company. memberlist[5] is our implementation of SWIM, which underpins Consul[6], a popular service discovery and management tool, and Nomad[7], a high-availability, data center scale scheduler. Through our relationship with customers, we know of hundreds of thousands of running instances of Consul, and deployments with more than 6,000 members in a single group.

REFERENCES

- [1] A. Das, I. Gupta, and A. Motivala, “SWIM: Scalable Weakly-consistent Infection-style Process Group Membership Protocol,” in *Proceedings of the 2002 International Conference on Dependable Systems and Networks*, ser. DSN ’02.
- [2] (2017, Nov.) Butterfly documentation. Chef. [Online]. Available: <https://www.habitat.sh/docs/internals/#supervisor-internals>
- [3] (2017, Nov.) Habitat. [Online]. Available: <https://www.habitat.sh>
- [4] (2017, Nov.) Ringpop documentation. Uber Technologies Inc. [Online]. Available: https://ringpop.readthedocs.io/en/latest/architecture_design.html
- [5] HashiCorp, “memberlist Project,” <https://github.com/hashicorp/memberlist>, 2017, [Online; accessed 28-Feb-2018].
- [6] ——, “Consul Project,” <https://www.consul.io/>, 2017, [Online; accessed 28-Feb-2018].
- [7] ——, “Nomad Project,” <https://www.nomadproject.io/>, 2017, [Online; accessed 28-Feb-2018].
- [8] P. Huang, C. Guo, L. Zhou, J. R. Lorch, Y. Dang, M. Chintalapati, and R. Yao, “Gray failure: The achilles’ heel of cloud-scale systems,” in *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*, ser. HotOS ’17.
- [9] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, “Epidemic algorithms for replicated database maintenance,” in *Proceedings of the Sixth Annual ACM Symposium on Principles of Distributed Computing*, 1987.
- [10] A. Dadgar, J. Phillips, and J. Currey, “Lifeguard : Swim-ing with local health awareness,” vol. abs/1707.00788, 2017. [Online]. Available: <http://arxiv.org/abs/1707.00788>
- [11] T. D. Chandra and S. Toueg, “Unreliable failure detectors for reliable distributed systems,” *J. ACM*.
- [12] W. Chen, S. Toueg, and M. K. Aguilera, “On the quality of service of

References Create "The Citation Graph"



Papers == nodes

References == edges

**c.f. web graph's
pages and URLs**

Problem: References Only Look Back



References are directed edges, and following them only let us go back in time...

"paper X references (prior) papers A, B and C"

Solution: The Transpose Graph



Reverses every edge, so "X references A" becomes "A is referenced by X"

Now we can look forward in time ...

"paper A is referenced by (subsequent) papers X, Y and Z"

Bi-Directional Graph Traversal



Combining citation graph and its transpose is powerful

From an interesting paper, can go back to look at its references ... and also check subsequent work that fixed, improved or built on it.

Google Scholar Quietly Does This



An introduction to computational networks and the computational network toolkit (Invited Talk)
D Yu, A Eversole, M Rossbach, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
Fifteenth Annual International Conference on Grid and Utility Computing, GUC'13, May 2013, Paris, France

Dandelion: a distributed system for distributed storage and processing of large-scale datasets
CJ Rossbach, M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
Proceedings of the Fifteenth Annual International Conference on Grid and Utility Computing, GUC'13, May 2013, Paris, France

Supporting I/O requests in distributed systems
J Currey, S Bakshi, M Isard, V Prabhakaran, U Wieder, K Talwar, E Saada
Systems for Future Internet, IFI 2013, Paris, France

PTask: operating system support for distributed computing clusters
CJ Rossbach, M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
Proceedings of the Fifteenth Annual International Conference on Grid and Utility Computing, GUC'13, May 2013, Paris, France

Operating System Support for MapReduce
CJ Rossbach, M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
HotOS 13, 32-35 June 2013, Berkeley, CA, USA

Quincy: fair scheduling for distributed computing clusters
M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles

DryadLINQ: a distributed system for parallel computation with fine-grained resource sharing
YYMID Fetterly, M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada
Proc. LSDS-IR 2013, 10-11 October 2013, Berkeley, CA, USA

Some samples of our work
Y Yu, M Isard, J Currey, U Wieder, K Talwar, E Saada
Tech. Rep. MS-TR-13-001, Microsoft Research, Redmond, WA, USA

System and method for scheduling concurrent jobs on clusters
KR Black, RM Riedel, M Isard, J Currey, U Wieder, K Talwar, E Saada
US Patent 7,016,911, 2006

System and method for scheduling concurrent jobs on clusters
JJ Currey, JE Riedel, KR Black, M Isard, U Wieder, K Talwar, E Saada
US Patent 7,761,230, 2010

Wireless Access Point Selection for Quality of Service
K Black, J Currey, M Isard, V Prabhakaran, U Wieder, K Talwar, E Saada
OMG Meeting in San Jose, CA, USA

Real-time computation on distributed resources
J Currey, M Isard, V Prabhakaran, U Wieder, K Talwar, E Saada
Embedded Systems Conference San Jose, CA, USA

Michael Isard
Research Scientist, Google
Amaz...
d Vice ...
ience, ...
ech & ...
nion, I...
olumbia U...
Universi...

Quincy: fair scheduling for distributed computing clusters
Michael Isard, Vijayan Prabhakaran, Jon Currey, Udi Wieder, Kunal Talwar, Andrew Goldberg
Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles, 2009, pp. 261-276

Publication date: 2009/10/11
Conference: Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles
Pages: 261-276
Publisher: ACM
Description: This paper addresses the problem of scheduling concurrent jobs on clusters where application data is stored on the computing nodes. This setting, in which scheduling computations close to their data is crucial for performance, is increasingly common and arises in systems such as MapReduce, Hadoop, and Dryad as well as many grid-computing environments. We argue that data-intensive computation benefits from a fine-grain resource sharing model that differs from the coarser semi-static resource allocations implemented by most existing cluster computing architectures. The problem of scheduling with locality and fairness constraints has not previously been extensively studied under this resource-sharing model.

We introduce a powerful and flexible new framework for scheduling concurrent distributed jobs with fine-grain resource sharing. The scheduling problem is mapped to a graph datastructure, where edge ...

Total citations: Cited by 856

Year	Total Citations
2010	~10
2011	~20
2012	~30
2013	~40
2014	~50
2015	~60
2016	~50
2017	~40
2018	~50
2019	~10

Scholar articles: Quincy: fair scheduling for distributed computing clusters
M Isard, V Prabhakaran, J Currey, U Wieder, K Talwar, E Saada - Proceedings of the ACM SIGOPS 22nd symposium on ..., 2009
Cited by 856 Related articles All 17 versions

Subsequent Work Is Just A Click Away



Articles About 856 results (0.03 sec)

Any time
Since 2019
Since 2018
Since 2015
Custom range...

Sort by relevance
 Sort by date

include citations
 Create alert

Quincy: fair scheduling for distributed computing clusters
 Search within citing articles

[PDF] **Improving MapReduce performance in heterogeneous environments.** [PDF] usenix.org
M Zaharia, A Konwinski, AD Joseph, RH Katz, I Stoica - OsdI, 2008 - static.usenix.org
MapReduce is emerging as an important programming model for large-scale data-parallel applications such as web indexing, data mining, and scientific simulation. Hadoop is an open-source implementation of MapReduce enjoying wide adoption and is often used for ...
☆ 99 Cited by 1837 Related articles All 38 versions ☰

[PDF] **Mesos: A platform for fine-grained resource sharing in the data center.** [PDF] usenix.org
B Hindman, A Konwinski, M Zaharia, A Ghodsi... - NSDI, 2011 - static.usenix.org
We present Mesos, a platform for sharing commodity clusters between multiple diverse cluster computing frameworks, such as Hadoop and MPI. Sharing improves cluster utilization and avoids per-framework data replication. Mesos shares resources in a fine ...
☆ 99 Cited by 1468 Related articles All 47 versions ☰

[PDF] **Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling** [PDF] psu.edu
M Zaharia, D Borthakur, J Sen Sarma... - Proceedings of the 5th ..., 2010 - dl.acm.org
As organizations start to use data-intensive cluster computing systems like Hadoop and Dryad for more applications, there is a growing need to share clusters between users. However, there is a conflict between fairness in scheduling and data locality (placing tasks ...
☆ 99 Cited by 1425 Related articles All 18 versions ☰

[PDF] **Dominant Resource Fairness: Fair Allocation of Multiple Resource Types.** [PDF] usenix.org
A Ghodsi, M Zaharia, B Hindman, A Konwinski... - Nsdi, 2011 - static.usenix.org
We consider the problem of fair resource allocation in a system containing different resource types, where each user may have different demands for each resource. To address this problem, we propose Dominant Resource Fairness (DRF), a generalization of max-min ...
☆ 99 Cited by 889 Related articles All 37 versions ☰

[PDF] **Reining in the Outliers in Map-Reduce Clusters using Mantri.** [PDF] usenix.org
G Ananthanarayanan, S Kandula, AG Greenberg... - OsdI, 2010 - usenix.org
Experience from an operational Map-Reduce cluster reveals that outliers significantly prolong job completion. e causes for outliers include run-time contention for processor, memory and other resources, disk failures, varying bandwidth and congestion along network ...
☆ 99 Cited by 688 Related articles All 26 versions ☰



Other Research Discovery Methods

Recommendation Systems

Curated Collections

Social Graph : Online and Real-Life



Recommendation Systems

Google Scholar



Jon Currey
HashiCorp Research
Verified email at hashicorp.com
Distributed Systems and M...



Google Scholar

Delay scheduling: a simple technique for achieving locality and fairness in clu 

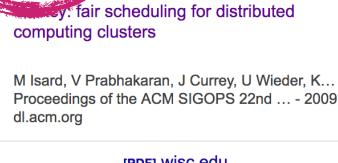
Articles

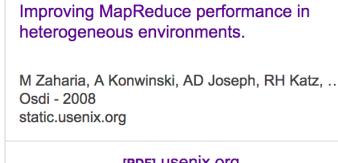
Any time
Since 2019
Since 2018
Since 2015
Custom range...

Sort by relevance
Sort by date

 include patents
 include citations

 Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling
M Zaharia, D Borthakur, J Sen Sarma... - Proceedings of the 5th ..., 2010 - dl.acm.org
As organizations start to use data-intensive cluster computing systems like Hadoop and Dryad for more applications, there is a growing need to share clusters between users. However, there is a conflict between fairness in scheduling and data locality (placing tasks on nodes that contain their input data). We illustrate this problem through our experience designing a fair scheduler for a 600-node Hadoop cluster at Facebook. To address the conflict between locality and fairness, we propose a simple algorithm called delay ...
★ 99 Cited by 1425 Related articles All 18 versions 


M Isard, V Prabhakaran, J Currey, U Wieder, K... - Proceedings of the ACM SIGOPS 22nd ... - 2009 - dl.acm.org



M Zaharia, A Konwinski, AD Joseph, RH Katz, ... - Osd - 2008 - static.usenix.org



J Dean, S Ghemawat - Communications of the ACM - 2008 - dl.acm.org


Showing the best result for this search. See all results



Recommendation Systems

Arxiv Sanity Preserver

Built in spare time by @karpathy to accelerate research.
Serving last 77180 papers from cs.[CV|CL|LG|AI|NE]/stat.ML

bosconi | log out

Fork me on GitHub



quincy fair scheduling

most recent

top recent

top hype

friends

discussions

recommended

library

Toward a Standard Interface for User-Defined Scheduling in OpenMP

Vivek Kale, Christian Iwainsky, Michael Klemm, Jonas Kondorfer, Florina Ciorba

6/21/2019 cs.DC | cs.CL | cs.PF | cs.PL

15 pages with references

1906.08911v1 pdf

[show similar](#) | [discuss](#)



Downloaded from arxiv.com by [REDACTED] on [REDACTED]

Parallel loops are an important part of OpenMP programs. Efficient scheduling of parallel loops can improve performance of the programs. The current OpenMP specification only offers three options for loop scheduling, which are insufficient in certain instances. Given the large number of other possible scheduling strategies, it is infeasible to standardize each one. A more viable approach is to extend the OpenMP standard to allow for users to define loop scheduling strategies. The approach will enable standard-compliant application-specific scheduling. This work analyzes the principal components required by user-defined scheduling and proposes two competing interfaces as candidates for the OpenMP standard. We conceptually compare the two proposed interfaces with respect to the three host languages of OpenMP, i.e., C, C++, and Fortran. These interfaces serve the OpenMP community as a basis for discussion and prototype implementation for user-defined scheduling.



Curated Collections

[papers-we-love / papers-we-love](#) Watch ▾ 2,742 Unstar 35,191 Fork 3,320

[Code](#) [Issues 9](#) [Pull requests 4](#) [Wiki](#) [Security](#) [Insights](#)

Papers from the computer science community to read and discuss. <http://papersthatimread.com/>

[computer-science](#) [read-papers](#) [meetup](#) [papers](#) [programming](#) [theory](#) [awesome](#)

760 commits 1 branch 0 releases 199 contributors

Branch: master ▾ [New pull request](#) [Create new file](#) [Upload files](#) [Find File](#) [Clone or download ▾](#)

Author	Commit Message	Date
	kuskmen and hakutsuru A structure paper I come by every now and then. Useful for competitiv...	Latest commit 3dd27b0 17 days ago
	change to v8 since v1 is a buggy redirect loop (#531)	8 months ago
	Added Picard first paper on affective computing (#507)	last year
	added space after a md formatted link	5 years ago
	Update to all READMEs for hosted content	4 years ago
	Add: Artificial Intelligence - A Universal Music Translation Network (#...	2 months ago
	Adding the Marsyas open source audio processing framework. (#492)	2 years ago
	List 'A DNA-Based Archival Storage System' (#427)	3 years ago



Curated Collections

the morning paper

a random walk through Computer Science research, by Adrian Colyer

[ABOUT](#) [ARCHIVES](#) [INFOQ QR EDITIONS](#) [SEARCH](#) [SUBSCRIBE](#) [TAGS](#) [PRIVACY](#)

One SQL to rule them all: an efficient and syntactically idiomatic approach to management of streams and tables

JULY 3, 2019 ~ 2 COMMENTS

One SQL to rule them all: an efficient and syntactically idiomatic approach to management of streams and tables Begoli et al., SIGMOD'19 In data



Social Graph as Recommender

Follow people on Twitter who post relevant research

Attend Papers We Love and other Meetups

Conferences - Online and In Person



**Online Proceedings
Collections of new work
Curated by review**

**Attend
Social recommendations**



Why consume research?

How? (without breaking master)

Produce research?

Title here



**Body slide
3 line max.**

Research Problem Selection



Is it actually a research problem?

Do we have an insight?

Risk/reward profile

Problem Selection Inputs



Observed user or dev team challenges

Upcoming technology trends

External research results, new and old

Many Formats, Same Key Components



Author(s)

- Order is specific to publication community

Title

Publication venue

- Conference proceedings
- Journal
- ...
- 'pre-print'
- Not (yet) peer reviewed or formally published
- **arXiv.org** ("archive dot org")

REFERENCES

- [1] A. Das, I. Gupta, and A. Motivala, "SWIM: Scalable Weakly-consistent Infection-style Process Group Membership Protocol," in *Proceedings of the 2002 International Conference on Dependable Systems and Networks*, ser. DSN '02.
- [2] (2017, Nov.) Butterfly documentation. Chef. [Online]. Available: <https://www.habitat.sh/docs/internals/#supervisor-internals>
- [3] (2017, Nov.) Habitat. [Online]. Available: <https://www.habitat.sh>
- [4] (2017, Nov.) Ringpop documentation. Uber Technologies Inc. [Online]. Available: https://ringpop.readthedocs.io/en/latest/architecture_design.html
- [5] HashiCorp, "memberlist Project," <https://github.com/hashicorp/memberlist>, 2017, [Online; accessed 28-Feb-2018].
- [6] ——, "Consul Project," <https://www.consul.io/>, 2017, [Online; accessed 28-Feb-2018].
- [7] ——, "Nomad Project," <https://www.nomadproject.io/>, 2017, [Online; accessed 28-Feb-2018].
- [8] P. Huang, C. Guo, L. Zhou, J. R. Lorch, Y. Dang, M. Chintalapati, and R. Yao, "Gray failure: The achilles' heel of cloud-scale systems," in *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*, ser. HotOS '17.
- [9] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry, "Epidemic algorithms for replicated database maintenance," in *Proceedings of the Sixth Annual ACM Symposium on Principles of Distributed Computing*, 1987.
- [10] A. Dadgar, J. Phillips, and J. Currey, "Lifeguard : Swim-ing with local health awareness," vol. abs/1707.00788, 2017. [Online]. Available: <http://arxiv.org/abs/1707.00788>
- [11] T. D. Chandra and S. Toueg, "Unreliable failure detectors for reliable distributed systems," *J. ACM*.
- [12] W. Chen, S. Toueg, and M. K. Aguilera, "On the quality of service of



"The citation graph is one of humankind's most important intellectual achievements"

*Dario Taraborelli,
Director of Research, the Wikimedia Foundation*

Price (Science, Vol 149, 1965)



Networks of Scientific Papers

The pattern of bibliographic references indicates the nature of the scientific research front.

Derek J. de Solla Price

This article is an attempt to describe in the broadest outline the nature of the total world network of scientific papers. We shall try to picture the network which is obtained by linking each published paper to the other papers directly associated with it. To do this, let us consider that special relationship which is given by the citation of one paper by another in its footnotes or bibliography. I should make it clear, however, that this broad picture tells us something about the papers them-

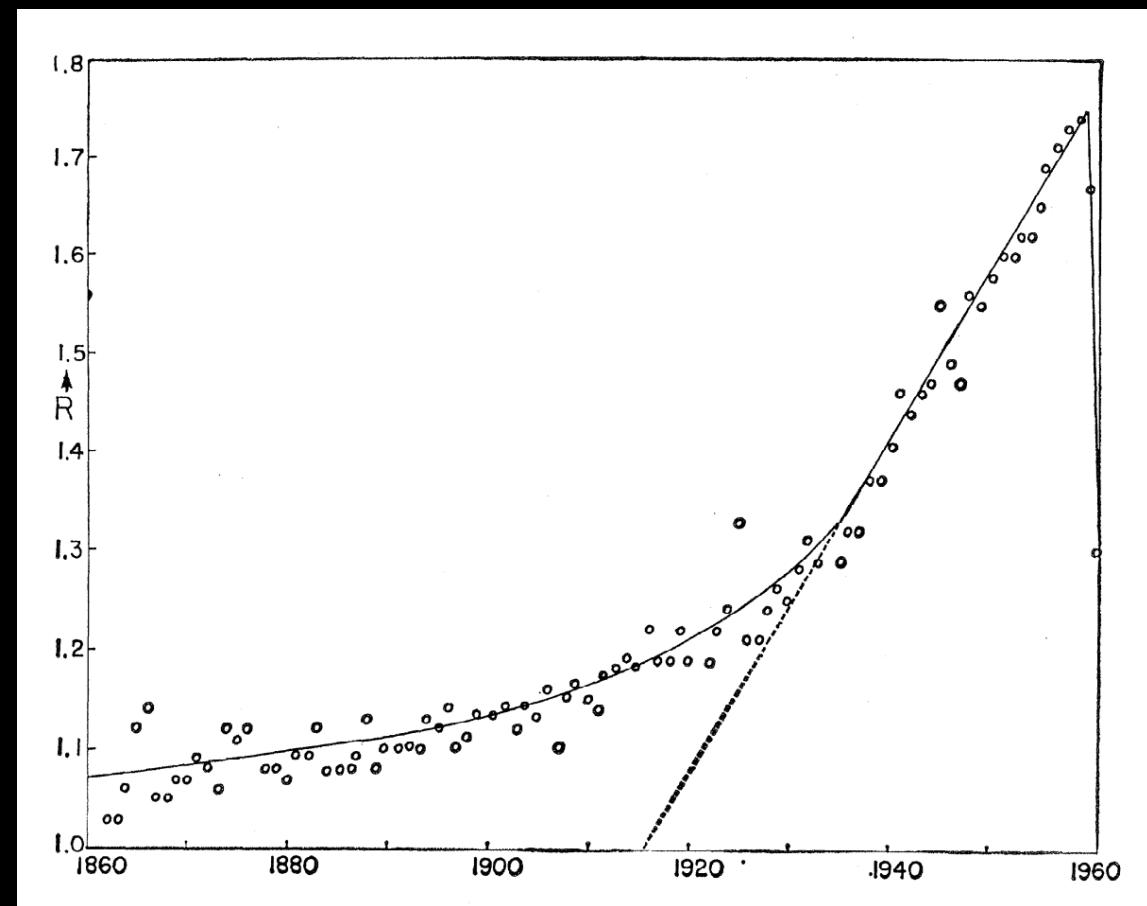
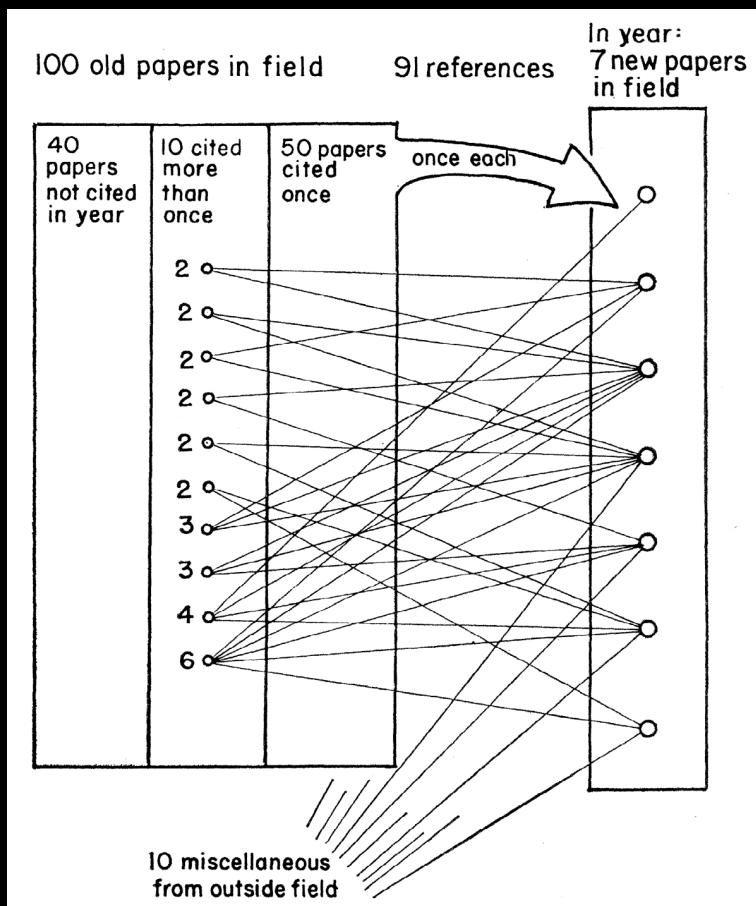
chine-handled citation studies, of large and representative portions of literature, which are much more tractable for such analysis than any topical indexing known to me. It is from such studies, by Garfield (1, 2), Kessler (3), Tukey (4), Osgood (5), and others, that I have taken the source data of this study.

Incidence of References

percent of the papers contain no references at all; this notwithstanding, 50 percent of the references come from the 85 percent of the papers that are of the "normal" research type and contain 25 or fewer references apiece. The distribution here is fairly flat; indeed about 5 percent of the papers fall in each of the categories of 3, 4, 5, 6, 7, 8, 9, and 10 references each. At the other end of the scale, there are review-type papers with many references each. About 25 percent of all references come from the 5 percent (of all papers) that contain 45 or more references each and average 75 to a paper, while 12 percent of the references come from the "fattest" category—the 1 percent (of all papers) that have 84 or more references each and average about 170 to a paper. It is interesting to note that the number of papers with n references falls off in this "fattest" category as $1/n^2$, up to many hundreds per paper.

These references, of course, cover the entire previous body of literature. We can calculate roughly that, since the body of world literature has been growing exponentially for a few centuries (6), and probably will continue at its present rate of growth of about 7

1965 Graph Visualization and Analysis



Knowledge is Power



Businesses Built on Access Control Politics, Activism and Reform

But How Do We Actually 'Do' Research? ↗

Research Problem Selection

Research Project Methodology

Research/Product Relationship

Relevant to Your Team Too?



Why *consume* research in engineering?

How to integrate into your dev process

Why and how to *produce* research



Title here



**Body slide
3 line max.**

Title here



**Body slide
3 line max.**