

# *Big Data on EC2: Mashing Technology in the Cloud*

**ShareThis**

*Paco Nathan, Data Insights Team*

# Scenario...

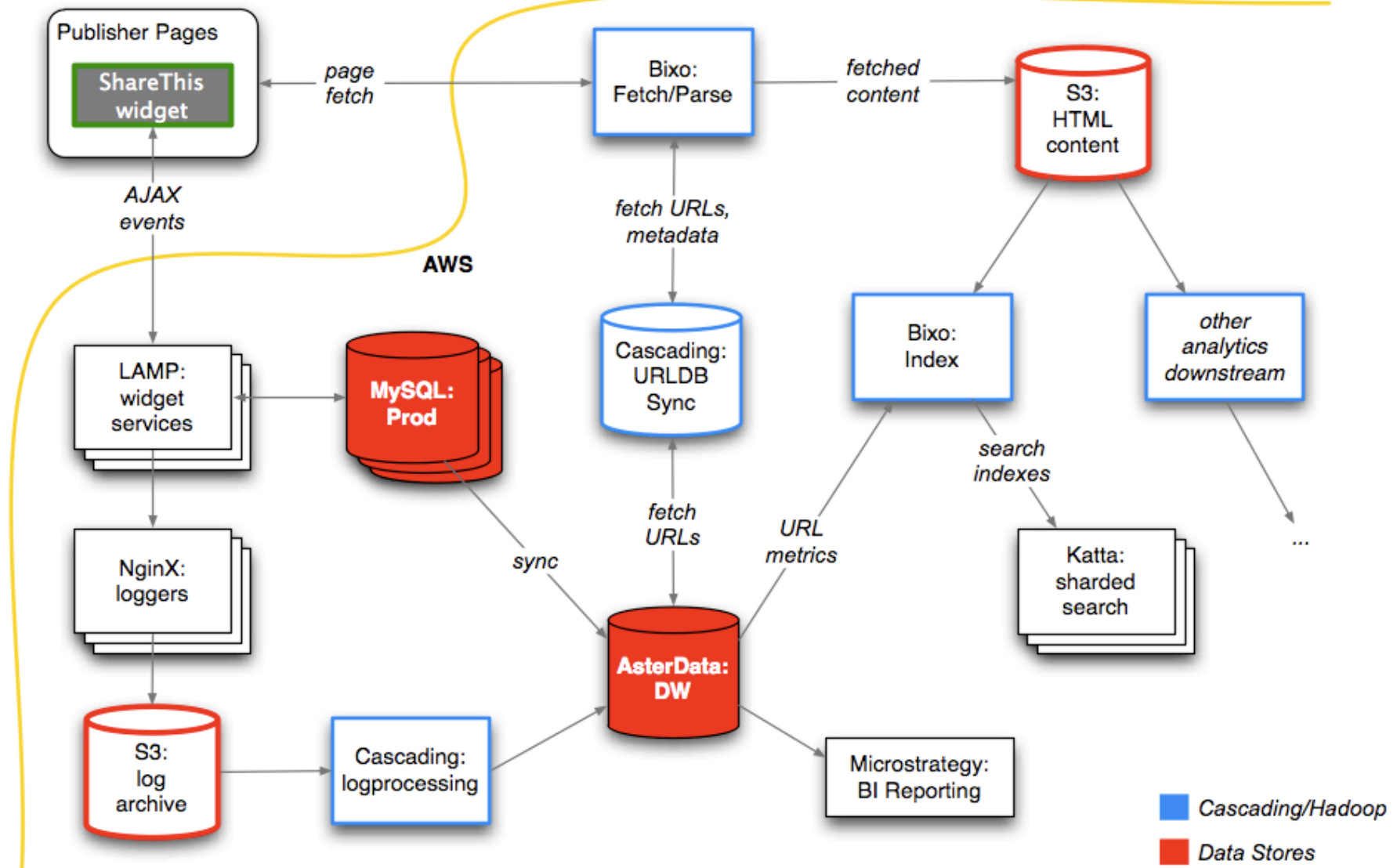
---

- ❖ Given:  $>10^6$  publishers,  $>10^9$  users,  $>10^{10}$  urls
- ❖ Early-stage start-up,  $< 25$  people, seeking to minimize spend on Ops and capex for data centers
- ❖ Serving widgets on page views for popular online publications: ESPN, HuffPost, FOX, CS Monitor, CBS Marketwatch, Wired, TechCrunch, etc.
- ❖ Spikes in popularity of stories leads to elastic demands throughout the system architecture: serving API, logging, DW, BI, etc.
- ❖ Business needs to improve user experience by analyzing how people share online content

# System Architecture

---

- ❖ 100% infrastructure based on Amazon AWS
- ❖ Each component designed for cost-effective, horizontal scale-out
- ❖ *AsterData*: infrastructure based on a “hub-and-spoke” pattern of batch jobs and data consolidation
- ❖ *Cascading*: abstraction layer for tying together system components
- ❖ Batch jobs run on Amazon *Elastic MapReduce* and AsterData *SQL/MR*
- ❖ Vertical search based on *Bixo* and *Katta*



# Cascading

---

- ❖ Syntax is for humans, APIs are for software
- ❖ Cascading defines apps in terms of functions applied to data flows, incorporates end-points; whereas MR is coarse, key/values are brittle
- ❖ Direct benefits to team's responsibilities, process, deliverables
- ❖ Also consider impact on team size, composition, staffing, training
- ❖ Ideal for provisioning/monitoring elastic resources, such as EMR
- ❖ Great potential to allow for migrating apps
- ❖ (see also: Cascading talk @ Hadoop Summit tomorrow, 6pm)

# Amazon *Elastic MapReduce*

---

- ❖ Can scale-out wide and select different instance types at launch
- ❖ Reads input from S3, writes output to S3, optionally copies logs to S3
- ❖ Excellent command line tools make dev/test/debug cycle more efficient
- ❖ Risks for DIY Hadoop clusters: time+cost to acquire leases, data locality, etc., — EMR resolves these to make large apps more cost-effective
- ❖ Simplifies needs for Ops — which can be quite difficult and expensive to staff for Big Data apps, and pose a risk to scale-out strategy
- ❖ See the Cascading examples: *LogAnalyzer for CloudFront* and *Multitool*

# AsterData *nCluster Cloud Edition*

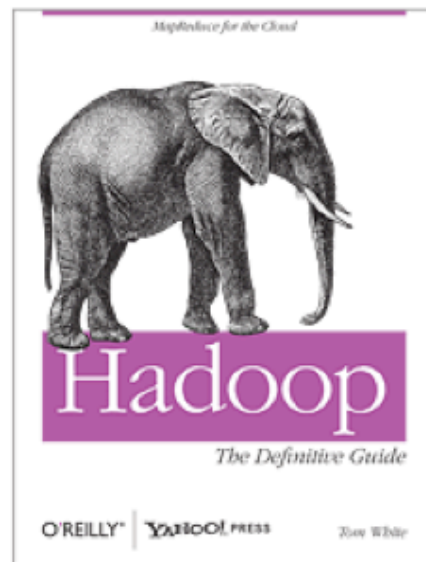
---

- ❖ Scalable, fault-tolerant, relational database — directly in the cloud
- ❖ Takes only minutes to go from idea to a prototype which scales well; accessible by data scientists who have less coding background
- ❖ Use of SQL unions on map phase, plus other SQL primitives for shuffle and reduce phases, allows for complex MR workflows
- ❖ Leveraging SQL primitives during shuffle and reduce phases allows for automated optimizations
- ❖ Contributed code among developer community, implementing libraries for popular algorithms based on In-Database MapReduce

# ShareThis as Case Study

---

See also: *ShareThis case study, "Cascading"*  
by Chris K Wensel, in...



## **Hadoop: The Definitive Guide** **MapReduce for the Cloud**

**by Tom White**  
**June 2009**

**<http://hadoopbook.com>**

*"A comprehensive resource for using Hadoop to build reliable, scalable, distributed systems. Programmers will find details for analyzing large datasets with Hadoop, and administrators will learn how to set up and run Hadoop clusters. The book includes case studies that illustrate how Hadoop is used to solve specific problems."*

# Contacts:

<http://sharethis.com>

[@pacoid](#) on Twitter

<http://cascading.org>

[http://asterdata.com/product/ncluster\\_cloud.php](http://asterdata.com/product/ncluster_cloud.php)

<http://aws.amazon.com/elasticmapreduce>

<http://github.com/emi/bixo>

<http://katta.sourceforge.net>

<http://www.hadoopbook.com>