**Data**

**Courtesy :** [**zenkay**](https://github.com/zenkay)**/[bigdata-ecosystem](https://github.com/zenkay/bigdata-ecosystem)**

**Projects**

* [Frameworks](https://github.com/zenkay/bigdata-ecosystem#frameworks)
* [Distributed Programming](https://github.com/zenkay/bigdata-ecosystem#distributed-programming)
* [Distributed Filesystem](https://github.com/zenkay/bigdata-ecosystem#distributed-filesystem)
* [Key-Map Data Model](https://github.com/zenkay/bigdata-ecosystem#key-map-data-model)
* [Document Data Model](https://github.com/zenkay/bigdata-ecosystem#document-data-model)
* [Key-value Data Model](https://github.com/zenkay/bigdata-ecosystem#key-value-data-model)
* [Graph Data Model](https://github.com/zenkay/bigdata-ecosystem#graph-data-model)
* [NewSQL Databases](https://github.com/zenkay/bigdata-ecosystem#newsql-databases)
* [Columnar Databases](https://github.com/zenkay/bigdata-ecosystem#columnar-databases)
* [Time-Series Databases](https://github.com/zenkay/bigdata-ecosystem#time-series-databases)
* [SQL-like processing](https://github.com/zenkay/bigdata-ecosystem#sql-like-processing)
* [Integrated Development Environments](https://github.com/zenkay/bigdata-ecosystem#integrated-development-environments)
* [Data Ingestion](https://github.com/zenkay/bigdata-ecosystem#data-ingestion)
* [Message-oriented middleware](https://github.com/zenkay/bigdata-ecosystem#message-oriented-middleware)
* [Service Programming](https://github.com/zenkay/bigdata-ecosystem#service-programming)
* [Scheduling](https://github.com/zenkay/bigdata-ecosystem#scheduling)
* [Machine Learning](https://github.com/zenkay/bigdata-ecosystem#machine-learning)
* [Benchmarking](https://github.com/zenkay/bigdata-ecosystem#benchmarking)
* [Security](https://github.com/zenkay/bigdata-ecosystem#security)
* [System Deployment](https://github.com/zenkay/bigdata-ecosystem#system-deployment)
* [Container Manager](https://github.com/zenkay/bigdata-ecosystem#container-manager)
* [Applications](https://github.com/zenkay/bigdata-ecosystem#applications)
* [Search engine and framework](https://github.com/zenkay/bigdata-ecosystem#search-engine-and-framework)
* [MySQL forks and evolutions](https://github.com/zenkay/bigdata-ecosystem#mysql-forks-and-evolutions)
* [PostgreSQL forks and evolutions](https://github.com/zenkay/bigdata-ecosystem#postgresql-forks-and-evolutions)
* [Memcached forks and evolutions](https://github.com/zenkay/bigdata-ecosystem#memcached-forks-and-evolutions)
* [Embedded Databases](https://github.com/zenkay/bigdata-ecosystem#embedded-databases)
* [Business Intelligence](https://github.com/zenkay/bigdata-ecosystem#business-intelligence)
* [Data Analysis](https://github.com/zenkay/bigdata-ecosystem#data-analysis)
* [Data Warehouse](https://github.com/zenkay/bigdata-ecosystem#data-warehouse)
* [Data Visualization](https://github.com/zenkay/bigdata-ecosystem#data-visualization)
* [Internet of Things](https://github.com/zenkay/bigdata-ecosystem#internet-of-things)

**Frameworks**

* [Apache Hadoop](http://hadoop.apache.org/) - framework for distributed processing. Integrates MapReduce (parallel processing), YARN (job scheduling) and HDFS (distributed file system).

**Distributed Programming**

* [AddThis Hydra](https://github.com/addthis/hydra) - distributed data processing and storage system originally developed at AddThis.
* [Akela](https://github.com/mozilla-metrics/akela) - Mozilla's utility library for Hadoop, HBase, Pig, etc..
* [Amazon Lambda](http://aws.amazon.com/lambda/) - a compute service that runs your code in response to events and automatically manages the compute resources for you.
* [Amazon SPICE](http://www.datanami.com/2015/10/07/amazon-debuts-fast-cheap-bi-with-quicksight/) - Super-fast Parallel In-memory Calculation Engine.
* [AMPcrowd](http://amplab.github.io/ampcrowd/) - A RESTful web service that runs microtasks across multiple crowds.
* [AMPLab G-OLA](https://amplab.cs.berkeley.edu/publication/g-ola-generalized-online-aggregation-for-interactive-analysis-on-big-data/) - a novel mini-batch execution model that generalizes OLA to support general OLAP queries with arbitrarily nested aggregates using efficient delta maintenance techniques.
* [AMPLab SIMR](http://databricks.github.io/simr/) - run Spark on Hadoop MapReduce v1.
* [Apache Crunch](http://crunch.apache.org/) - a simple Java API for tasks like joining and data aggregation that are tedious to implement on plain MapReduce.
* [Apache DataFu](http://incubator.apache.org/projects/datafu.html) - collection of user-defined functions for Hadoop and Pig developed by LinkedIn.
* [Apache Flink](http://flink.incubator.apache.org/) - high-performance runtime, and automatic program optimization.
* [Apache Gora](http://gora.apache.org/) - framework for in-memory data model and persistence.
* [Apache Hama](http://hama.apache.org/) - BSP (Bulk Synchronous Parallel) computing framework.
* [Apache Ignite](http://ignite.apache.org/) - high-performance, integrated and distributed in-memory platform for computing and transacting on large-scale data sets in real-time.
* [Apache MapReduce](http://wiki.apache.org/hadoop/MapReduce/) - programming model for processing large data sets with a parallel, distributed algorithm on a cluster.
* [Apache Pig](https://pig.apache.org/) - high level language to express data analysis programs for Hadoop.
* [Apache S4](http://incubator.apache.org/s4/) - framework for stream processing, implementation of S4.
* [Apache Spark](http://spark.incubator.apache.org/) - framework for in-memory cluster computing.
* [Apache Spark Streaming](http://spark.incubator.apache.org/docs/0.7.3/streaming-programming-guide.html) - framework for stream processing, part of Spark.
* [Apache Storm](http://storm-project.net/) - framework for stream processing by Twitter also on YARN.
* [Apache Tez](http://tez.incubator.apache.org/) - application framework for executing a complex DAG (directed acyclic graph) of tasks, built on YARN.
* [Apache Twill](https://incubator.apache.org/projects/twill.html) - abstraction over YARN that reduces the complexity of developing distributed applications.
* [Arvados](https://arvados.org/) - Spins a web of microservices around unsuspecting sysadmins.
* [Blaze](http://blaze.pydata.org/en/latest/) - Python users high-level access to efficient computation on inconveniently large data.
* [Cascalog](http://cascalog.org/) - data processing and querying library.
* [Cheetah](http://vldbarc.org/pvldb/vldb2010/pvldb_vol3/I08.pdf) - High Performance, Custom Data Warehouse on Top of MapReduce.
* [Concurrent Cascading](http://www.cascading.org/) - framework for data management/analytics on Hadoop.
* [Damballa Parkour](https://github.com/damballa/parkour) - MapReduce library for Clojure.
* [Datasalt Pangool](https://github.com/datasalt/pangool) - alternative MapReduce paradigm.
* [DataTorrent StrAM](https://www.datatorrent.com/) - real-time engine is designed to enable distributed, asynchronous, real time in-memory big-data computations in as unblocked a way as possible, with minimal overhead and impact on performance.
* [DistributedR](http://www.vertica.com/distributedr/) - scalable high-performance platform for the R language.
* [Drools](http://www.drools.org/) - a Business Rules Management System (BRMS) solution.
* [eBay Oink](https://github.com/eBay/oink) - REST based interface for PIG execution.
* [Esper](http://esper.codehaus.org/) - a highly scalable, memory-efficient, in-memory computing, SQL-standard, minimal latency, real-time streaming-capable Big Data processing engine for historical data.
* [Facebook Corona](https://www.facebook.com/notes/facebook-engineering/under-the-hood-scheduling-mapreduce-jobs-more-efficiently-with-corona/10151142560538920) - Hadoop enhancement which removes single point of failure.
* [Facebook Peregrine](http://peregrine_mapreduce.bitbucket.org/) - Map Reduce framework.
* [Facebook Scuba](https://www.facebook.com/notes/facebook-engineering/under-the-hood-data-diving-with-scuba/10150599692628920) - distributed in-memory datastore.
* [GearPump](https://github.com/intel-hadoop/gearpump) - a lightweight real-time big data streaming engine.
* [Geotrellis](http://geotrellis.io/) - geographic data processing engine for high performance applications.
* [GetStream Stream Framework](https://github.com/tschellenbach/Stream-Framework) - a Python library, which allows you to build newsfeed and notification systems using Cassandra and/or Redis.
* [GIS Tools for Hadoop](http://esri.github.io/gis-tools-for-hadoop/) - Big Data Spatial Analytics for the Hadoop Framework.
* [Google Dataflow](http://googledevelopers.blogspot.it/2014/06/cloud-platform-at-google-io-new-big.html) - create data pipelines to help themæingest, transform and analyze data.
* [Google FlumeJava](http://research.google.com/pubs/pub35650.html) - Easy, Efficient Data-Parallel Pipelines. Base of Google Dataflow.
* [Google MapReduce](http://research.google.com/archive/mapreduce.html) - map reduce framework.
* [Google MillWheel](http://research.google.com/pubs/pub41378.html) - fault tolerant stream processing framework.
* [GraphLab Dato](https://dato.com/products/create/open_source.html) - fast, scalable engine of GraphLab Create, a Python library.
* [Hazelcast](http://hazelcast.com/products/hazelcast/) - In-Memory Data Grid.
* [HParser](http://www.informatica.com/us/products/big-data/hparser/) - data parsing transformation environment optimized for Hadoop.
* [IBM Streams](http://www.ibm.com/software/products/en/infosphere-streams) - advanced analytic platform that allows user-developed applications to quickly ingest, analyze and correlate information as it arrives from thousands of real-time sources.
* [JAQL](https://code.google.com/p/jaql/) - declarative programming language for working with structured, semi-structured and unstructured data.
* [Kite](http://kitesdk.org/docs/current/) - is a set of libraries, tools, examples, and documentation focused on making it easier to build systems on top of the Hadoop ecosystem.
* [Kryo](https://github.com/EsotericSoftware/kryo) - Java serialization and cloning: fast, efficient, automatic.
* [LinkedIn Cubert](https://github.com/linkedin/Cubert) - a fast and efficient batch computation engine for complex analysis and reporting of massive datasets on Hadoop.
* [Lipstick](https://github.com/Netflix/Lipstick) - Pig workflow visualization tool.
* [Metamarkers Druid](http://druid.io/) - framework for real-time analysis of large datasets.
* [Microsoft Azure Stream Analytics](http://azure.microsoft.com/en-us/services/stream-analytics/) - an event processing engine that helps uncover real-time insights from devices, sensors, infrastructure, applications and data.
* [Microsoft Orleans](http://research.microsoft.com/en-us/projects/orleans/) - a straightforward approach to building distributed high-scale computing applications.
* [Microsoft Project Orleans](http://dotnet.github.io/orleans/) - a framework that provides a straightforward approach to building distributed high-scale computing applications.
* [Microsoft Trill](http://research.microsoft.com/en-us/projects/trill/) - a high-performance in-memory incremental analytics engine.
* [Netflix Aegisthus](https://github.com/Netflix/aegisthus) - Bulk Data Pipeline out of Cassandra. implements a reader for the SSTable format and provides a map/reduce program to create a compacted snapshot of the data contained in a column family.
* [Netflix Lipstick](https://github.com/Netflix/Lipstick) - Pig Visualization framework.
* [Netflix Mantis](http://qconsf.com/presentation/mantis-netflixs-event-stream-processing-system) - Event Stream Processing System.
* [Netflix PigPen](https://github.com/Netflix/PigPen) - map-reduce for Clojure whiche compiles to Apache Pig.
* [Netflix STAASH](https://github.com/Netflix/staash) - language-agnostic as well as storage-agnostic web interface for storing data into persistent storage systems.
* [Netflix Surus](https://github.com/Netflix/Surus) - a collection of tools for analysis in Pig and Hive.
* [Netflix Zeno](https://github.com/Netflix/zeno) - Netflix's In-Memory Data Propagation Framework.
* [Nextflow](http://www.nextflow.io/) - Dataflow oriented toolkit for parallel and distributed computational pipelines.
* [Nokia Disco](http://discoproject.org/) - MapReduce framework developed by Nokia.
* [Oryx](http://oryx.io/) - is a realization of the lambda architecture built on Apache Spark and Apache Kafka, but with specialization for real-time large scale machine learning.
* [Pachyderm](http://pachyderm.io/) - lets you store and analyze your data using containers..
* [Parsely Streamparse](https://github.com/Parsely/streamparse) - streamparse lets you run Python code against real-time streams of data. It also integrates Python smoothly with Apache Storm..
* [PigPen](https://github.com/Netflix/PigPen) - PigPen is map-reduce for Clojure, or distributed Clojure. It compiles to Apache Pig, but you don't need to know much about Pig to use it.
* [Pinterest Pinlater](http://engineering.pinterest.com/post/91288882494/pinlater-an-asynchronous-job-execution-system) - asynchronous job execution system.
* [Pubnub](http://www.pubnub.com/) - Data stream network.
* [Pydoop](http://pydoop.sourceforge.net/docs/) - Python MapReduce and HDFS API for Hadoop.
* [ScaleOut hServer](http://www.scaleoutsoftware.com/) - fast, scalable in-memory data grid for Hadoop.
* [SeqPig](http://seqpig.sourceforge.net/) - Simple and scalable scripting for large sequencing data set(ex: bioinfomation) in Hadoop .
* [SigmoidAnalytics Spork](https://github.com/sigmoidanalytics/spork) - Pig on Apache Spark.
* [SNAP](https://github.com/snap-stanford/snap) - Stanford Network Analysis Platform is a general purpose, high performance system for analysis and manipulation of large networks.
* [spark-dataflow](https://github.com/cloudera/spark-dataflow) - allows users to execute dataflow pipelines with Spark.
* [SpatialHadoop](http://spatialhadoop.cs.umn.edu/) - SpatialHadoop is a MapReduce extension to Apache Hadoop designed specially to work with spatial data. .
* [Spring for Apache Hadoop](http://projects.spring.io/spring-hadoop/) - unified configuration model and easy to use APIs for using HDFS, MapReduce, Pig, and Hive.
* [SQLStream Blaze](http://www.sqlstream.com/blaze/) - stream processing platform.
* [Stratio Crossdata](http://docs.stratio.com/modules/crossdata/development) - provides an unified way to access to multiple datastores.
* [Stratio Decision](http://docs.stratio.com/modules/decision/development/) - the union of a real-time messaging bus with a complex event processing engine using Spark Streaming.
* [Stratio Streaming](http://www.openstratio.org/about/stratio-streaming/) - the union of a real-time messaging bus with a complex event processing engine using Spark Streaming.
* [Stratosphere](http://stratosphere.eu/) - general purpose cluster computing framework.
* [Streamdrill](https://streamdrill.com/) - usefull for counting activities of event streams over different time windows and finding the most active one.
* [Succinct Spark](http://succinct.cs.berkeley.edu/wp/wordpress/) - Enabling Queries on Compressed Data.
* [Sumo Logic](http://www.sumologic.com/) - cloud based analyzer for machine-generated data..
* [Teradata QueryGrid](http://it.teradata.com/Teradata-QueryGrid/) - data-access layer that can orchestrate multiple modes of analysis across multiple databases plus Hadoop.
* [TIBCO ActiveSpaces](http://www.tibco.com/products/automation/in-memory-computing/in-memory-data-grid/activespaces-enterprise-edition) - in-memory data grid.
* [Tigon](http://cask.co/products/tigon/) - a distributed framework built on Apache HadoopTM and Apache HBaseTM for real-time, high-throughput, low-latency data processing and analytics applications.
* [Torch](http://torch.ch/) - Scientific computing for LuaJIT.
* [Trident](https://storm.apache.org/documentation/Trident-tutorial.html) - a high-level abstraction for doing realtime computing on top of Storm.
* [Twitter Crane](http://www.theregister.co.uk/2010/06/29/twitter_to_open_source_crane/) - Java ETL.
* [Twitter Gizzard](https://github.com/twitter/gizzard) - a flexible sharding framework for creating eventually-consistent distributed datastores.
* [Twitter Heron](http://twitter.github.io/heron/) - a realtime, distributed, fault-tolerant stream processing engine from Twitter.
* [Twitter Scalding](https://github.com/twitter/scalding) - Scala library for Map Reduce jobs, built on Cascading.
* [Twitter Summingbird](https://github.com/twitter/summingbird) - Streaming MapReduce with Scalding and Storm, by Twitter.
* [Twitter TSAR](https://blog.twitter.com/2014/tsar-a-timeseries-aggregator) - TimeSeries AggregatoR by Twitter.

**Distributed Filesystem**

* [Amazon Elastic File System](https://aws.amazon.com/efs/) - file storage service for Amazon Elastic Compute Cloud (Amazon EC2) instances.
* [Amazon Simple Storage Service](http://aws.amazon.com/s3/) - secure, durable, highly-scalable object storage.
* [Apache HDFS](http://hadoop.apache.org/) - a way to store large files across multiple machines.
* [Apache Kudu](http://getkudu.io/) - completes Hadoop's storage layer to enable fast analytics on fast data.
* [BeeGFS](http://www.fhgfs.com/cms/) - formerly FhGFS, parallel distributed file system.
* [Ceph Filesystem](http://ceph.com/ceph-storage/file-system/) - software storage platform designed.
* [Disco DDFS](http://disco.readthedocs.org/en/latest/howto/ddfs.html) - distributed filesystem.
* [Facebook Haystack](https://www.facebook.com/note.php?note_id=76191543919) - object storage system.
* [Google Cloud Storage](https://cloud.google.com/storage/) - durable and highly available object storage.
* [Google Cloud Storage Nearline](https://cloud.google.com/storage-nearline/) - a highly available, affordable solution for backup, archiving and disaster recovery..
* [Google Colossus](https://google.com/) - distributed filesystem (GFS2).
* [Google GFS](https://google.com/) - distributed filesystem.
* [Google Megastore](http://research.google.com/pubs/pub36971.html) - scalable, highly available storage.
* [GridGain](http://www.gridgain.org/) - GGFS, Hadoop compliant in-memory file system.
* [HDSF-DU](https://github.com/twitter/hdfs-du) - HDFS-DU is an interactive visualization of the Hadoop distributed file system. .
* [Lustre file system](http://wiki.lustre.org/) - high-performance distributed filesystem.
* [MapR-FS](http://doc.mapr.com/display/MapR/MapR-FS) - Distributed filesystem from MapR.
* [Microsoft Azure Data Lake](http://azure.microsoft.com/en-us/campaigns/data-lake/) - a hyper scale repository for big data analytic workloads.
* [Netflix S3mper](https://github.com/Netflix/s3mper) - library that provides an additional layer of consistency checking on top of Amazon's S3 index through use of a consistent, secondary index.
* [Quantcast File System QFS](https://www.quantcast.com/engineering/qfs/) - open-source distributed file system.
* [Red Hat GlusterFS](http://www.gluster.org/) - scale-out network-attached storage file system.
* [Tachyon](http://tachyon-project.org/) - reliable file sharing at memory speed across cluster frameworks.

**Key-Map Data Model**

* [Actian Vector](http://www.actian.com/) - column-oriented analytic database.
* [Apache Accumulo](http://accumulo.apache.org/) - distribuited key/value store, built on Hadoop.
* [Apache Cassandra](http://cassandra.apache.org/) - column-oriented distribuited datastore, inspired by BigTable.
* [Apache HBase](http://hbase.apache.org/) - column-oriented distribuited datastore, inspired by BigTable.
* [Facebook HydraBase](https://code.facebook.com/posts/321111638043166/hydrabase-the-evolution-of-hbase-facebook/) - evolution of HBase made by Facebook.
* [Google BigTable](http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/archive/bigtable-osdi06.pdf) - column-oriented distributed datastore.
* [Google Cloud Datastore](https://developers.google.com/datastore/) - is a fully managed, schemaless database for storing non-relational data over BigTable.
* [Hypertable](http://hypertable.org/) - column-oriented distribuited datastore, inspired by BigTable.
* [InfiniDB](http://infinidb.co/) - is accessed through a MySQL interface and use massive parallel processing to parallelize queries.
* [MapR-DB](http://content.dataversity.net/rs/wilshireconferences/images/MapR-DB_Product_Preview_for_NoSQL_Now.pdf) - fast, scalable, and enterprise-ready in-Hadoop database architected to manage big data.
* [Netflix Priam](https://github.com/Netflix/Priam) - Co-Process for backup/recovery, Token Management, and Centralized Configuration management for Cassandra.
* [OhmData C5](http://ohmdata.com/) - improved version of HBase.
* [Palantir AtlasDB](https://www.palantir.com/2014/06/atlasdb-transactions-for-distributed-key-value-stores-part-i/) - a massively scalable datastore and transactional layer that can be placed on top of any key-value store to give it ACID properties.
* [Sqrrl](http://sqrrl.com/product/sqrrl-enterprise/) - NoSQL databases on top of Apache Accumulo.
* [Stratio Cassandra](http://docs.stratio.com/modules/cassandra-lucene-index/development/) - Cassandra index functionality has been extended to provide near real time search such as ElasticSearch or Solr, including full text search capabilities and multivariable, geospatial and bitemporal search.
* [Tephra](https://github.com/continuuity/tephra) - Transactions for HBase.
* [Twitter Manhattan](https://blog.twitter.com/2014/manhattan-our-real-time-multi-tenant-distributed-database-for-twitter-scale) - real-time, multi-tenant distributed database for Twitter scale.

**Document Data Model**

* [Actian Versant](http://www.actian.com/products/operational-databases/) - commercial object-oriented database management systems .
* [Amazon SimpleDB](http://aws.amazon.com/simpledb/) - a highly available and flexible non-relational data store that offloads the work of database administration.
* [BigchainDB](https://www.bigchaindb.com/) - The scalable blockchain database..
* [Clusterpoint](http://www.clusterpoint.com/) - a database software for high-speed storage and large-scale processing of XML and JSON data on clusters of commodity hardware.
* [Crate Data](https://crate.io/) - is an open source massively scalable data store. It requires zero administration.
* [Facebook Apollo](http://www.infoq.com/news/2014/06/facebook-apollo) - Facebook’s Paxos-like NoSQL database.
* [jumboDB](http://comsysto.github.io/jumbodb/) - document oriented datastore over Hadoop.
* [LinkedIn Ambry](https://github.com/linkedin/ambry) - Distributed object store.
* [LinkedIn Espresso](http://data.linkedin.com/projects/espresso) - horizontally scalable document-oriented NoSQL data store.
* [MarkLogic](http://www.marklogic.com/) - Schema-agnostic Enterprise NoSQL database technology.
* [Microsoft DocumentDB](http://azure.microsoft.com/en-us/services/documentdb/) - fully-managed, highly-scalable, NoSQL document database service.
* [Microsoft StorSimple](http://azure.microsoft.com/en-us/services/storsimple/) - a unique hybrid cloud storage solution that lowers costs and improves data protection.
* [MongoDB](http://www.mongodb.org/) - Document-oriented database system.
* [RavenDB](http://www.ravendb.net/) - A transactional, open-source Document Database.
* [RethinkDB](http://www.rethinkdb.com/) - document database that supports queries like table joins and group by.
* [Terrastore](https://code.google.com/p/terrastore/) - a modern document store which provides advanced scalability and elasticity features without sacrificing consistency.
* [TokuMX](http://www.tokutek.com/products/tokumx-for-mongodb/) - High-Performance MongoDB Distribution.
* [Tokutek](https://www.percona.com/) - Tokutek claims to improve MongoDB performance 20x.

**Key-value Data Model**

* [Aerospike](http://www.aerospike.com/) - NoSQL flash-optimized, in-memory. Open source and "Server code in 'C' (not Java or Erlang) precisely tuned to avoid context switching and memory copies..
* [Amazon DynamoDB](http://aws.amazon.com/dynamodb/) - distributed key/value store, implementation of Dynamo paper.
* [Couchbase ForestDB](https://github.com/couchbaselabs/forestdb) - Fast Key-Value Storage Engine Based on Hierarchical B+-Tree Trie.
* [Edis](http://inaka.github.io/edis/) - is a protocol-compatible Server replacement for Redis.
* [ElephantDB](https://github.com/nathanmarz/elephantdb) - Distributed database specialized in exporting data from Hadoop.
* [EventStore](http://geteventstore.com/) - distributed time series database.
* [Exasolution](http://www.exasol.com/en/products/exasolution/) - an in-memory, column-oriented, relational database management system.
* [HyperDex](http://hyperdex.org/) - next generation key-value store.
* [KAI](http://sourceforge.net/projects/kai/) - a distributed key-value datastore.
* [LinkedIn Krati](https://github.com/linkedin-sna/sna-page/tree/master/krati) - is a simple persistent data store with very low latency and high throughput.
* [Linkedin Voldemort](http://www.project-voldemort.com/voldemort/) - distributed key/value storage system.
* [MemcacheDB](http://memcachedb.org/) - a distributed key-value storage system designed for persistent.
* [Netflix Dynomite](http://techblog.netflix.com/2014/03/netflixoss-season-2-episode-1.html) - thin Dynamo-based replication for cached data.
* [Oracle NoSQL Database](http://www.oracle.com/technetwork/database/database-technologies/nosqldb/overview/index.html) - distributed key-value database by Oracle Corporation.
* [QDB](https://github.com/reborndb/qdb) - A fast, high availability, fully Redis compatible store.
* [RAMCloud](https://ramcloud.atlassian.net/wiki/display/RAM/RAMCloud) - storage system that provides large-scale low-latency storage by keeping all data in DRAM all the time and aggregating the main memories of thousands of servers.
* [RebornDB](https://github.com/reborndb/reborn) - Distributed database fully compatible with redis protocol.
* [Redis](http://redis.io/) - in memory key value datastore.
* [Redis Cluster](http://redis.io/topics/cluster-spec) - distributed implementation of Redis.
* [Redis Sentinel](http://redis.io/topics/sentinel) - system designed to help managing Redis instances.
* [Riak](https://github.com/basho/riak) - a decentralized datastore.
* [Scalaris](https://code.google.com/p/scalaris/) - a distributed transactional key-value store.
* [Storehaus](https://github.com/twitter/storehaus) - library to work with asynchronous key value stores, by Twitter.
* [Tarantool](https://github.com/tarantool/tarantool) - an efficient NoSQL database and a Lua application server.
* [TreodeDB](https://github.com/Treode/store) - key-value store that's replicated and sharded and provides atomic multirow writes.
* [Yahoo Sherpa](http://en.wikipedia.org/wiki/Yahoo_Sherpa) - hosted, distributed and geographically replicated key-valueÊcloud storage platform.

**Graph Data Model**

* [Apache Giraph](http://giraph.apache.org/) - implementation of Pregel, based on Hadoop.
* [Apache Spark Bagel](http://spark.incubator.apache.org/docs/0.7.3/bagel-programming-guide.html) - implementation of Pregel, part of Spark.
* [ArangoDB](https://www.arangodb.org/) - multi model distribuited database.
* [Doradus](https://github.com/dell-oss/Doradus) - Doradus is a REST service that extends a Cassandra NoSQL database with a graph-based data model, advanced indexing and search features, and a REST API.
* [Facebook TAO](https://www.facebook.com/notes/facebook-engineering/tao-the-power-of-the-graph/10151525983993920) - TAO is the distributed data store that is widely used at facebook to store and serve the social graph.
* [Faunus](http://thinkaurelius.github.io/faunus/) - Hadoop-based graph analytics engine for analyzing graphs represented across a multi-machine compute cluster.
* [Google Cayley](https://github.com/google/cayley) - open-source graph database.
* [Google Pregel](http://kowshik.github.io/JPregel/pregel_paper.pdf) - graph processing framework.
* [GraphLab PowerGraph](http://graphlab.org/projects/source.html) - a core C++ GraphLab API and a collection of high-performance machine learning and data mining toolkits built on top of the GraphLab API.
* [GraphX](https://amplab.cs.berkeley.edu/publication/graphx-grades/) - resilient Distributed Graph System on Spark.
* [Gremlin](https://github.com/tinkerpop/gremlin) - graph traversal Language.
* [HyperGraphDB](http://www.hypergraphdb.org/) - general purpose, open-source data storage mechanism based on a powerful knowledge management formalism known as directed hypergraphs.
* [InfiniteGraph](http://www.objectivity.com/infinitegraph) - distributed graph database.
* [Infovore](https://github.com/paulhoule/infovore) - RDF-centric Map/Reduce framework.
* [Intel GraphBuilder](https://01.org/graphbuilder/) - tools to construct large-scale graphs on top of Hadoop.
* [MapGraph](http://mapgraph.io/) - Massively Parallel Graph processing on GPUs.
* [Mazerunner for Neo4j](https://github.com/kbastani/neo4j-mazerunner) - extends a Neo4j graph database to run scheduled big data graph compute algorithms at scale with HDFS and Apache Spark..
* [MemGraph](http://www.memgraph.io/) - cypher compatibile, high-performance in-memory transactional and real-time analytics graph database.
* [Microsoft Graph Engine](http://www.graphengine.io/) - a distributed, in-memory, large graph processing engine, underpinned by a strongly-typed RAM store and a general computation engine.
* [Neo4j](http://www.neo4j.org/) - graph database writting entirely in Java.
* [OrientDB](http://www.orientechnologies.com/) - document and graph database.
* [Phoebus](https://github.com/xslogic/phoebus) - framework for large scale graph processing.
* [Pinterest Zen](https://www.youtube.com/watch?v=yI0vHfgK6oI) - Pinterest's Graph Storage Service.
* [Sparksee](http://www.sparsity-technologies.com/) - scalable high-performance graph database.
* [Stardog](http://stardog.com/) - graph database: search, query, reasoning, and constraints in a lightweight, pure Java system.
* [Titan](http://thinkaurelius.github.io/titan/) - distributed graph database, built over Cassandra.
* [Twitter FlockDB](https://github.com/twitter/flockdb) - distribuited graph database.

**NewSQL Databases**

* [Actian Ingres](http://www.actian.com/products/operational-databases/) - commercially supported, open-source SQL relational database management system.
* [BayesDB](http://probcomp.csail.mit.edu/bayesdb/index.html) - statistic oriented SQL database.
* [Cockroach](https://github.com/cockroachdb/cockroach) - Scalable, Geo-Replicated, Transactional Datastore.
* [Datomic](http://www.datomic.com/) - distributed database designed to enable scalable, flexible and intelligent applications.
* [FoundationDB](https://foundationdb.com/) - distributed database, inspired by F1.
* [Google F1](http://research.google.com/pubs/pub41344.html) - distributed SQL database built on Spanner.
* [Google Spanner](http://research.google.com/archive/spanner.html) - globally distributed semi-relational database.
* [H-Store](http://hstore.cs.brown.edu/) - is an experimental main-memory, parallel database management system that is optimized for on-line transaction processing (OLTP) applications.
* [HandlerSocket](http://www.percona.com/doc/percona-server/5.5/performance/handlersocket.html) - NoSQL plugin for MySQL/MariaDB.
* [IBM DB2](http://www.ibm.com/software/data/db2/) - object-relational database management system.
* [InfiniSQL](http://www.infinisql.org/) - infinity scalable RDBMS.
* [MemSQL](http://www.memsql.com/) - in memory SQL database witho optimized columnar storage on flash.
* [NuoDB](http://www.nuodb.com/) - SQL/ACID compliant distributed database.
* [Oracle Database](http://www.oracle.com/us/corporate/features/database-12c/index.html) - object-relational database management system.
* [Oracle TimesTen in-Memory Database](http://www.oracle.com/technetwork/database/database-technologies/timesten/overview/index.html) - in-memory, relational database management system with persistence and recoverability.
* [Pivotal GemFire XD](http://gemfirexd.docs.gopivotal.com/latest/userguide/index.html?q=about_users_guide.html/) - Low-latency, in-memory, distributed SQL data store. Provides SQL interface to in-memory table data, persistable in HDFS.
* [SAP HANA](http://www.saphana.com/welcome) - is an in-memory, column-oriented, relational database management system.
* [Segment SQL](https://segment.com/redshift) - Track your customer data to Amazon Redshift.
* [SenseiDB](http://senseidb.com/) - distributed, realtime, semi-structured database.
* [Sky](http://skydb.io/) - database used for flexible, high performance analysis of behavioral data.
* [SymmetricDS](http://www.symmetricds.org/) - open source software for both file and database synchronization.
* [Teradata Database](http://it.teradata.com/products-and-services/Teradata-Database/) - complete relational database management system.
* [VoltDB](http://voltdb.com/) - in-memory NewSQL database.

**Columnar Databases**

* [Amazon RedShift](http://aws.amazon.com/redshift/) - data warehouse service, based on PostgreSQL.
* [Apache Arrow](http://arrow.apache.org/) - Powering Columnar In-Memory Analytics.
* [C-Store](http://db.lcs.mit.edu/projects/cstore/) - column oriented DBMS.
* [Google BigQuery](http://research.google.com/pubs/pub36632.html) - framework for interactive analysis, implementation of Dremel.
* [Google Dremel](http://research.google.com/pubs/pub36632.html) - framework for interactive analysis, implementation of Dremel.
* [MonetDB](https://www.monetdb.org/) - column store database.
* [Parquet](http://parquet.io/) - columnar storage format for Hadoop.
* [Pivotal Greenplum](https://www.pivotal.io/big-data/pivotal-greenplum-database) - purpose-built, dedicated analytic data warehouse.
* [Vertica](http://www.vertica.com/) - is designed to manage large, fast-growing volumes of data and provide very fast query performance when used for data warehouses.

**Time-Series Databases**

* [Chronix](http://www.chronix.io/) - fast and efficient time series storage based on Apache Lucene and Apache Solr.
* [Cube](http://square.github.io/cube/) - uses MongoDB to store time series data.
* [Etsy StatsD](https://github.com/etsy/statsd/) - simple daemon for easy stats aggregation.
* [InfluxDB](http://influxdb.com/) - distributed time series database.
* [Kairos](https://pypi.python.org/pypi/kairos) - Time series data storage in Redis, Mongo, SQL and Cassandra.
* [Kairosdb](https://code.google.com/p/kairosdb/) - similar to OpenTSDB but allows for Cassandra.
* [OpenTSDB](http://opentsdb.net/) - distributed time series database on top of HBase.
* [Prometheus](http://prometheus.io/) - an open-source service monitoring system and time series database.
* [Square Cube](http://square.github.io/cube/) - system for collecting timestamped events and deriving metrics.
* [TempoIQ](https://tempoiq.com/) - Cloud-based sensor analytics.

**SQL-like processing**

* [Actian SQL for Hadoop](http://www.actian.com/products/analytics-platform/) - high performance interactive SQL access to all Hadoop data.
* [Adabas](https://en.wikipedia.org/wiki/ADABAS) - ADABAS was NoSQL from a time when there was no SQL.
* [Akiban](http://www.akiban.com/) - Touted as SQL database with object structured storage.
* [AMPLAB Shark](https://github.com/amplab/shark/) - data warehouse system for Spark.
* [Apache Drill](http://incubator.apache.org/drill/) - framework for interactive analysis, inspired by Dremel.
* [Apache HCatalog](http://hive.apache.org/docs/hcat_r0.5.0/) - table and storage management layer for Hadoop.
* [Apache Hive](http://hive.apache.org/) - SQL-like data warehouse system for Hadoop.
* [Apache Optiq](https://wiki.apache.org/incubator/OptiqProposal) - framework that allows efficient translation of queries involving heterogeneous and federated data.
* [Apache Phoenix](http://phoenix.incubator.apache.org/index.html) - SQL skin over HBase.
* [BlinkDB](http://blinkdb.org/) - massively parallel, approximate query engine.
* [Brytlyt](http://www.brytlyt.co.uk/technology/) - a fully enabled GPGPU database which allows for offloading of database operations to General Processing on Graphics Processor Units..
* [Cloudera Impala](http://www.cloudera.com/content/cloudera/en/products-and-services/cdh/impala.html) - framework for interactive analysis, Inspired by Dremel.
* [Concurrent Lingual](http://www.cascading.org/lingual/) - SQL-like query language for Cascading.
* [Datasalt Splout SQL](http://www.datasalt.com/products/splout-sql/) - full SQL query engine for big datasets.
* [eBay Kylin](http://www.kylin.io/) - Distributed Analytics Engine from eBay Inc. that provides SQL interface and multi-dimensional analysis (OLAP) on Hadoop supporting extremely large datasets.
* [Facebook PrestoDB](http://prestodb.io/) - distributed SQL query engine.
* [Hadapt](http://hadapt.com/) - a native implementation of SQL for the Apache Hadoop open-source project.
* [Hekaton](https://en.wikipedia.org/wiki/Hekaton_(database)) - Refer to lock-free architecture for SQL Server 2014.
* [JethroData](http://jethrodata.com/product-2/product/) - index-based SQL engine for Hadoop.
* [Metanautix Quest](https://metanautix.com/product/) - data compute engine.
* [Pivotal HAWQ](http://www.gopivotal.com/pivotal-products/data/pivotal-hd) - SQL-like data warehouse system for Hadoop.
* [RainstorDB](http://rainstor.com/products/rainstor-database/) - database for storing petabyte-scale volumes of structured and semi-structured data.
* [Spark Catalyst](https://github.com/apache/spark/tree/master/sql) - is a Query Optimization Framework for Spark and Shark.
* [SparkSQL](http://databricks.com/blog/2014/03/26/Spark-SQL-manipulating-structured-data-using-Spark.html) - Manipulating Structured Data Using Spark.
* [Splice Machine](http://www.splicemachine.com/) - a full-featured SQL-on-Hadoop RDBMS with ACID transactions.
* [Stinger](http://hortonworks.com/labs/stinger/) - interactive query for Hive.
* [Tajo](http://tajo.incubator.apache.org/) - distributed data warehouse system on Hadoop.
* [Trafodion](https://wiki.trafodion.org/wiki/index.php/Main_Page) - enterprise-class SQL-on-HBase solution targeting big data transactional or operational workloads.

**Integrated Development Environments**

* [R-Studio](https://github.com/rstudio/rstudio) - IDE for R.

**Data Ingestion**

* [Amazon Kinesis](http://aws.amazon.com/kinesis/) - real-time processing of streaming data at massive scale.
* [Amazon Snowball](https://aws.amazon.com/importexport/) - a petabyte-scale data transport solution that uses secure appliances to transfer large amounts of data into and out of AWS.
* [AMPLab SampleClean](http://sampleclean.org/) - scalable techniques for data cleaning and statistical inference on dirty data.
* [Apache BookKeeper](http://zookeeper.apache.org/bookkeeper/) - a distributed logging service called BookKeeper and a distributed publish/subscribe system built on top of BookKeeper called Hedwig.
* [Apache Chukwa](http://incubator.apache.org/chukwa/) - data collection system.
* [Apache Flume](http://flume.apache.org/) - service to manage large amount of log data.
* [Apache Samza](http://samza.incubator.apache.org/) - stream processing framework, based on Kafla and YARN.
* [Apache Sqoop](http://sqoop.apache.org/) - tool to transfer data between Hadoop and a structured datastore.
* [Apache UIMA](https://uima.apache.org/) - Unstructured Information Management applications are software systems that analyze large volumes of unstructured information in order to discover knowledge that is relevant to an end user.
* [Cloudera Morphlines](https://github.com/cloudera/cdk/tree/master/cdk-morphlines) - framework that help ETL to Solr, HBase and HDFS.
* [Facebook Scribe](https://github.com/facebook/scribe) - streamed log data aggregator.
* [Fluentd](http://fluentd.org/) - tool to collect events and logs.
* [Google Photon](http://research.google.com/pubs/pub41318.html) - geographically distributed system for joining multiple continuously flowing streams of data in real-time with high scalability and low latency.
* [Heka](https://github.com/mozilla-services/heka) - open source stream processing software system.
* [HIHO](https://github.com/sonalgoyal/hiho) - framework for connecting disparate data sources with Hadoop.
* [LinkedIn Camus](https://github.com/linkedin/camus) - Kafka to HDFS pipeline. It is a mapreduce job that does distributed data loads out of Kafka.
* [LinkedIn Databus](http://data.linkedin.com/projects/databus) - stream of change capture events for a database.
* [LinkedIn Gobblin](http://engineering.linkedin.com/data-ingestion/gobblin-big-data-ease) - a framework for Solving Big Data Ingestion Problem.
* [LinkedIn Kamikaze](https://github.com/linkedin/kamikaze) - utility package for compressing sorted integer arrays.
* [Linkedin Lumos](http://www.slideshare.net/Hadoop_Summit/th-220p230-cramachandranv1) - bridge from OLTP to OLAP for use it on Hadoop.
* [LinkedIn White Elephant](https://github.com/linkedin/white-elephant) - log aggregator and dashboard.
* [Logstash](http://logstash.net/) - a tool for managing events and logs.
* [Netflix Ribbon](https://github.com/Netflix/ribbon) - a Inter Process Communication (remote procedure calls) library with built in software load balancers. The primary usage model involves REST calls with various serialization scheme support.
* [Netflix Suro](https://github.com/Netflix/suro) - data pipeline service for collecting, aggregating, and dispatching large volume of application events including log data based on Chukwa.
* [Pinterest Secor](https://github.com/pinterest/secor) - is a service implementing Kafka log persistance.
* [Record Breaker](http://cloudera.github.io/RecordBreaker/) - Automatic structure for your text-formatted data.
* [Sawmill](http://sawmill.net/) - extensive log processing and reporting features.
* [Stratio Ingestion](http://docs.stratio.com/modules/ingestion/development) - Apache Flume with steroids.
* [TIBCO Enterprise Message Service](http://www.tibco.com/products/automation/enterprise-messaging/enterprise-message-service) - standards-based messaging middleware.
* [Twitter Zipkin](https://github.com/twitter/zipkin) - distributed tracing system that helps us gather timing data for all the disparate services at Twitter.
* [Vibe Data Stream](http://www.informatica.com/us/products/big-data/vibe-data-stream/) - streaming data collection for real-time Big Data analytics.

**Message-oriented middleware**

* [ActiveMQ](http://activemq.apache.org/) - open source messaging and Integration Patterns server.
* [Amazon Simple Queue Service](http://aws.amazon.com/sqs/) - fast, reliable, scalable, fully managed queue service.
* [Apache Kafka](http://kafka.apache.org/) - distributed publish-subscribe messaging system.
* [Apache Qpid](http://qpid.apache.org/) - messaging tools that speak AMQP and support many languages and platforms.
* [Apcera NATS](http://nats.io/) - an open-source, high-performance, lightweight cloud native messaging system.
* [Apollo](http://activemq.apache.org/apollo/) - ActiveMQ's next generation of messaging.
* [Azure Event Hubs](http://azure.microsoft.com/en-us/services/event-hubs/) - a highly scalable publish-subscribe event ingestor.
* [Beanstalkd](http://kr.github.io/beanstalkd/) - simple, fast work queue.
* [Bit.ly NSQ](http://nsq.io/) - realtime distributed message processing at scale.
* [Celery](http://www.celeryproject.org/) - Distributed Task Queue.
* [Crossroads I/O](http://www.crossroads.io/) - library for building scalable and high performance distributed applications.
* [Darner](https://github.com/wavii/darner) - simple, lightweight message queue.
* [Facebook Iris](https://code.facebook.com/posts/820258981365363/building-mobile-first-infrastructure-for-messenger/) - a totally ordered queue of messaging updates with separate pointers into the queue indicating the last update sent to your Messenger app and the traditional storage tier.
* [Gearman](http://gearman.org/) - Job Server.
* [Google Cloud Pub/Sub](https://cloud.google.com/pubsub/) - reliable, many-to-many, asynchronous messaging hosted on Google's infrastructure.
* [Google Pub/Sub](https://cloud.google.com/pubsub/) - reliable, many-to-many, asynchronous messaging hosted on Google's infrastructure.
* [HornetQ](http://www.jboss.org/hornetq) - open source project to build a multi-protocol, embeddable, very high performance, clustered, asynchronous messaging system.
* [IronMQ](http://www.iron.io/mq) - easy-to-use highly available message queuing service.
* [Kestrel](http://robey.github.io/kestrel/) - distributed message queue system.
* [Marconi](https://wiki.openstack.org/wiki/Marconi) - queuing and notification service made by and for OpenStack, but not only for it.
* [RabbitMQ](http://www.rabbitmq.com/) - Robust messaging for applications.
* [RestMQ](http://restmq.com/) - message queue which uses HTTP as transport, JSON to format a minimalist protocol and is organized as REST resources.
* [RQ](http://python-rq.org/) - simple Python library for queueing jobs and processing them in the background with workers.
* [Sidekiq](http://sidekiq.org/) - Simple, efficient background processing for Ruby.
* [ZeroMQ](http://www.zeromq.org/) - The Intelligent Transport Layer.

**Service Programming**

* [Akka Toolkit](http://akka.io/) - runtime for distributed, and fault tolerant event-driven applications on the JVM.
* [Apache Avro](http://avro.apache.org/) - data serialization system.
* [Apache Curator](http://curator.apache.org/) - Java libaries for Apache ZooKeeper.
* [Apache Karaf](http://karaf.apache.org/) - OSGi runtime that runs on top of any OSGi framework.
* [Apache Thrift](http://thrift.apache.org/) - framework to build binary protocols.
* [Apache Zookeeper](http://zookeeper.apache.org/) - centralized service for process management.
* [Google Chubby](http://research.google.com/archive/chubby.html) - a lock service for loosely-coupled distributed systems.
* [Linkedin Norbert](http://data.linkedin.com/opensource/norbert) - cluster manager.
* [MPICH](http://www.mpich.org/) - high performance and widely portable implementation of the Message Passing Interface (MPI) standard.
* [OpenMPI](http://www.open-mpi.org/) - message passing framework.
* [Serf](http://www.serfdom.io/) - decentralized solution for service discovery and orchestration.
* [Spotify Luigi](https://github.com/spotify/luigi) - a Python package for building complex pipelines of batch jobs. It handles dependency resolution, workflow management, visualization, handling failures, command line integration, and much more.
* [Spring XD](https://github.com/spring-projects/spring-xd) - distributed and extensible system for data ingestion, real time analytics, batch processing, and data export.
* [Twitter Elephant Bird](https://github.com/kevinweil/elephant-bird) - libraries for working with LZOP-compressed data.
* [Twitter Finagle](https://twitter.github.io/finagle/) - asynchronous network stack for the JVM.

**Scheduling**

* [AirBnB Airflow](https://github.com/airbnb/airflow) - AirFlow is a system to programmatically author, schedule and monitor data pipelines.
* [Apache Aurora](http://aurora.incubator.apache.org/) - is a service scheduler that runs on top of Apache Mesos.
* [Apache Falcon](http://falcon.incubator.apache.org/) - data management framework.
* [Apache Oozie](http://oozie.apache.org/) - workflow job scheduler.
* [Chronos](http://airbnb.github.io/chronos/) - distributed and fault-tolerant scheduler.
* [Linkedin Azkaban](http://azkaban.github.io/azkaban2/) - batch workflow job scheduler.
* [Pinterest Pinball](http://engineering.pinterest.com/post/74429563460/pinball-building-workflow-management) - customizable platform for creating workflow managers.
* [Sparrow](https://github.com/radlab/sparrow) - scheduling platform.

**Machine Learning**

* [Amazon Machine Learning](https://aws.amazon.com/machine-learning/) - visualization tools and wizards that guide you through the process of creating machine learning (ML) models without having to learn complex ML algorithms and technology.
* [AMPLab Splash](http://zhangyuc.github.io/splash/) - a general framework for parallelizing stochastic learning algorithms on multi-node clusters.
* [AMPLab Velox](https://amplab.cs.berkeley.edu/projects/velox/) - a data management system for facilitating the next steps in real-world, large-scale analytics pipelines.
* [Apache Mahout](http://mahout.apache.org/) - machine learning library for Hadoop.
* [Ayasdi Core](http://www.ayasdi.com/) - tool for topological data analysis.
* [brain](https://github.com/harthur/brain) - Neural networks in JavaScript.
* [Caffe](http://caffe.berkeleyvision.org/) - a deep learning framework made with expression, speed, and modularity in mind. It is developed by the Berkeley Vision and Learning Cente.
* [Cloudera Oryx](https://github.com/cloudera/oryx) - real-time large-scale machine learning.
* [Concurrent Pattern](http://www.cascading.org/pattern/) - machine learning library for Cascading.
* [convnetjs](https://github.com/karpathy/convnetjs) - Deep Learning in Javascript. Train Convolutional Neural Networks (or ordinary ones) in your browser.
* [cuDNN](http://devblogs.nvidia.com/parallelforall/accelerate-machine-learning-cudnn-deep-neural-network-library/) - GPU-accelerated library of primitives for deep neural networks.
* [Decider](https://github.com/danielsdeleo/Decider) - Flexible and Extensible Machine Learning in Ruby.
* [DeepCL](https://github.com/hughperkins/DeepCL) - OpenCL library to train deep convolutional neural networks.
* [etcML](http://www.etcml.com/) - text classification with machine learning.
* [Etsy Conjecture](https://github.com/etsy/Conjecture) - scalable Machine Learning in Scalding.
* [Facebook DeepText](https://code.facebook.com/posts/181565595577955/introducing-deeptext-facebook-s-text-understanding-engine/) - a deep learning-based text understanding engine that can understand with near-human accuracy the textual content of several thousands posts per second, spanning more than 20 languages.
* [Facebook FBLearner Flow](https://code.facebook.com/posts/1072626246134461/introducing-fblearner-flow-facebook-s-ai-backbone/) - provides innovative functionality, like automatic generation of UI experiences from pipeline definitions and automatic parallelization of Python code using futures.
* [fbcunn](https://github.com/facebook/fbcunn) - Deep Learning CUDA Extensions from Facebook AI Research.
* [Google DistBelief](http://research.google.com/pubs/pub40565.html) - software framework that can utilize computing clusters with thousands of machines to train large models.
* [Google Sibyl](http://users.soe.ucsc.edu/~niejiazhong/slides/chandra.pdf) - System for Large Scale Machine Learning at Google.
* [Google TensorFlow](http://www.tensorflow.org/) - an Open Source Software Library for Machine Intelligence.
* [H2O](http://0xdata.github.io/h2o/) - statistical, machine learning and math runtime for Hadoop.
* [IBM Watson](http://www.ibm.com/smarterplanet/us/en/ibmwatson/) - cognitive computing system.
* [KeystoneML](https://github.com/amplab/keystone) - Simplifying robust end-to-end machine learning on Apache Spark.
* [LinkedIn FeatureFu](https://github.com/linkedin/FeatureFu) - contains a collection of library/tools for advanced feature engineering to derive features on top of other features, or convert a light weighted model into a feature.
* [LinkedIn ml-ease](https://github.com/linkedin/ml-ease) - ADMM based large scale logistic regression.
* [Microsoft Azure Machine Learning](https://studio.azureml.net/) - is built on the machine learning capabilities already available in several Microsoft products including Xbox and Bing and using predefined templates and workflows.
* [Microsoft CNTK](http://www.cntk.ai/) - Computational Network Toolkit.
* [MLbase](http://www.mlbase.org/) - distributed machine learning libraries for the BDAS stack.
* [MLPNeuralNet](https://github.com/nikolaypavlov/MLPNeuralNet) - Fast multilayer perceptron neural network library for iOS and Mac OS X.
* [Neon](https://github.com/NervanaSystems/neon) - a highly configurable deep learning framework.
* [nupic](https://github.com/numenta/nupic) - Numenta Platform for Intelligent Computing: a brain-inspired machine intelligence platform, and biologically accurate neural network based on cortical learning algorithms.
* [OpenAI Gym](https://github.com/openai/gym) - a toolkit for developing and comparing reinforcement learning algorithms.
* [PredictionIO](http://prediction.io/) - machine learning server buit on Hadoop, Mahout and Cascading.
* [scikit-learn](https://github.com/scikit-learn/scikit-learn) - scikit-learn: machine learning in Python.
* [Seldon](http://www.seldon.io/) - an open source predictive analytics platform based upon Spark, Kafka and Hadoop.
* [Spark MLlib](http://spark.apache.org/docs/0.9.0/mllib-guide.html) - a Spark implementation of some common machine learning (ML) functionality.
* [Sparkling Water](http://databricks.com/blog/2014/06/30/sparkling-water-h20-spark.html) - combine H2OÕs Machine Learning capabilities with the power of the Spark platform.
* [SparkNet](https://github.com/amplab/SparkNet) - Distributed Neural Networks for Spark.
* [Theano](http://deeplearning.net/software/theano/) - Python package for deep learning that can utilize NVIDIA's CUDA toolkit to run on the GPU.
* [Thunder](http://thefreemanlab.com/thunder/) - Large-scale analysis of neural data.
* [Vahara](https://github.com/Ganglion/varaha) - Machine learning and natural language processing with Apache Pig.
* [Velox](https://github.com/amplab/velox-modelserver) - a system for serving machine learning predictions.
* [Viv](http://viv.ai/) - global platform that enables developers to plug into and create an intelligent, conversational interface to anything.
* [Vowpal Wabbit](https://github.com/JohnLangford/vowpal_wabbit/wiki) - learning system sponsored by Microsoft and Yahoo!.
* [WEKA](http://www.cs.waikato.ac.nz/ml/weka/) - suite of machine learning software.
* [Wit](https://wit.ai/) - Natural Language for the Internet of Things.
* [Wolfram Alpha](http://www.wolframalpha.com/) - computational knowledge engine.
* [YHat ScienceOps](https://yhathq.com/products/scienceops) - platform for deploying, managing, and scaling predictive models in production applications.

**Benchmarking**

* [Apache Hadoop Benchmarking](https://issues.apache.org/jira/browse/MAPREDUCE-3561) - micro-benchmarks for testing Hadoop performances.
* [Berkeley SWIM Benchmark](https://github.com/SWIMProjectUCB/SWIM/wiki) - real-world big data workload benchmark.
* [Big-Bench](https://github.com/intel-hadoop/Big-Bench) - Big Bench Workload Development.
* [Hive-benchmarks](https://github.com/yhuai/hive-benchmarks) - some benchmarking queries for Apache Hive.
* [Hive-testbench](https://github.com/cartershanklin/hive-testbench) - Testbench for experimenting with Apache Hive at any data scale..
* [Intel HiBench](https://github.com/intel-hadoop/HiBench) - a Hadoop benchmark suite.
* [Mesosaurus](https://github.com/mesosphere/mesosaurus) - Mesos task load simulator framework for (cluster and Mesos) performance analysis.
* [Netflix Inviso](https://hadoopsummit.uservoice.com/forums/242807-hadoop-deployment-operations-track/suggestions/5568461-inviso-maximizing-big-data-performance-at-netflix) - performance focused Big Data tool.
* [PUMA Benchmarking](https://issues.apache.org/jira/browse/MAPREDUCE-5116) - benchmark suite for MapReduce applications.
* [Yahoo Gridmix3](https://developer.yahoo.com/blogs/hadoop/gridmix3-emulating-production-workload-apache-hadoop-450.html) - Hadoop cluster benchmarking from Yahoo engineer team.

**Security**

* [Apache Knox Gateway](http://knox.apache.org/) - single point of secure access for Hadoop clusters.
* [Apache Ranger](http://argus.incubator.apache.org/) - framework to enable, monitor and manage comprehensive data security across the Hadoop platform (formerly called Apache Argus).
* [Apache Sentry](http://incubator.apache.org/projects/sentry.html) - security module for data stored in Hadoop.
* [PacketPig](https://github.com/packetloop/packetpig) - Open Source Big Data Security Analytics.
* [Voltage SecureData](http://www.voltage.com/products/securedata-enterprise/) - data protection framework.

**System Deployment**

* [Ankush](https://github.com/impetus-opensource/ankush) - A big data cluster management tool that creates and manages clusters of different technologies..
* [Apache Ambari](http://ambari.apache.org/) - operational framework for Hadoop mangement.
* [Apache Bigtop](http://bigtop.apache.org/) - system deployment framework for the Hadoop ecosystem.
* [Apache Helix](http://helix.apache.org/) - cluster management framework.
* [Apache Mesos](http://mesos.apache.org/) - cluster manager.
* [Apache Slider](https://github.com/hortonworks/slider) - is a YARN application to deploy existing distributed applications on YARN.
* [Apache Whirr](http://whirr.apache.org/) - set of libraries for running cloud services.
* [Apache YARN](http://hortonworks.com/hadoop/yarn/) - Cluster manager.
* [Brooklyn](http://brooklyncentral.github.io/) - library that simplifies application deployment and management.
* [Buildoop](http://buildoop.github.io/) - Similar to Apache BigTop based on Groovy language.
* [Cloudera Director](http://www.cloudera.com/content/cloudera/en/products-and-services/director.html) - a comprehensive data management platform with the flexibility and power to evolve with your business.
* [Cloudera HUE](http://gethue.com/) - web application for interacting with Hadoop.
* [CloudPhysics](https://www.cloudphysics.com/product/) - collect operational metadata from your virtualized infrastructure, then correlate and analyze it to expose operational hazards and waste that pose a threat to your datacenter performance, efficiency and uptime.
* [Deimos](https://github.com/mesosphere/deimos) - Mesos containerizer hooks for Docker.
* [Develoop](http://deploop.github.io/) - tool for provisioning, managing and monitoring Apache Hadoop.
* [Etsy Sahale](https://codeascraft.com/2015/02/11/sahale-visualizing-cascading-workflows-at-etsy/) - Visualizing Cascading Workflows at Etsy.
* [Facebook Autoscale](https://code.facebook.com/posts/816473015039157/making-facebook-s-software-infrastructure-more-energy-efficient-with-autoscale/) - the load balancer will concentrate workload to a server until it has at least a medium-level workload.
* [Facebook Prism](http://www.wired.com/2012/08/facebook-prism/) - multi datacenters replication system.
* [Ganglia Monitoring System](http://ganglia.sourceforge.net/) - scalable distributed monitoring system for high-performance computing systems such as clusters and Grids.
* [Genie](https://github.com/Netflix/genie) - Genie provides REST-ful APIs to run Hadoop, Hive and Pig jobs, and to manage multiple Hadoop resources and perform job submissions across them..
* [Google Borg](http://www.wired.com/wiredenterprise/2013/03/google-borg-twitter-mesos/all/) - job scheduling and monitoring system.
* [Google Omega](https://www.youtube.com/watch?v=0ZFMlO98Jkc) - job scheduling and monitoring system.
* [Hannibal](https://github.com/sentric/hannibal) - Hannibal is tool to help monitor and maintain HBase-Clusters that are configured for manual splitting..
* [Hortonworks HOYA](http://hortonworks.com/blog/introducing-hoya-hbase-on-yarn/) - application that can deploy HBase cluster on YARN.
* [Jumbune](http://www.jumbune.org/) - Jumbune is an open-source product built for analyzing Hadoop cluster and MapReduce jobs..
* [Marathon](https://github.com/mesosphere/marathon) - Mesos framework for long-running services.
* [Minotaur](https://github.com/stealthly/minotaur) - scripts/recipes/configs to spin up VPC-based infrastructure in AWS from scratch and deploy labs to it.
* [Myriad](https://github.com/mesos/myriad) - a mesos framework designed for scaling YARN clusters on Mesos. Myriad can expand or shrink one or more YARN clusters in response to events as per configured rules and policies..
* [Neflix SimianArmy](https://github.com/Netflix/SimianArmy) - a suite of tools for keeping your cloud operating in top form.
* [Netflix Eureka](https://github.com/Netflix/eureka) - AWS Service registry for resilient mid-tier load balancing and failover.
* [Netflix Hystrix](https://github.com/Netflix/Hystrix) - a latency and fault tolerance library designed to isolate points of access to remote systems, services and 3rd party libraries, stop cascading failure and enable resilience in complex distributed systems where failure is inevitable.
* [Scaling Data](http://www.scalingdata.com/big-data) - tracing data center problems to root cause, predict capacity issues, identify emerging failures and highlight latent threats.
* [Stratio Manager](http://docs.stratio.com/modules/manager/development) - install, manage and monitor all the technology stack related to the Stratio Platform.
* [Tumblr Collins](http://tumblr.github.io/collins/) - Infrastructure management for engineers.
* [Tumblr Genesis](http://tumblr.github.io/genesis/) - a tool for data center automation.

**Container Manager**

* [Amazon EC2 Container Service](https://aws.amazon.com/ecs/) - a highly scalable, high performance container management service that supports Docker containers.
* [CoreOS Fleet](https://coreos.com/fleet/) - cluster management tool from CoreOS.
* [Docker](https://www.docker.com/) - an open platform for developers and sysadmins to build, ship, and run distributed applications.
* [Docker Swarm](https://docs.docker.com/swarm/) - native clustering for Docker.
* [Fig](http://www.fig.sh/) - fast, isolated development environments using Docker.
* [Google Container Engine](https://cloud.google.com/container-engine/) - Run Docker containers on Google Cloud Platform, powered by Kubernetes.
* [HashiCorp Nomad](https://www.nomadproject.io/) - a Distributed, Highly Available, Datacenter-Aware Scheduler.
* [Kubernetes](https://github.com/GoogleCloudPlatform/kubernetes) - open source implementation of container cluster management.
* [Pumba](https://github.com/gaia-adm/pumba) - Chaos testing tool for Docker.
* [Rocket](https://coreos.com/blog/rocket/) - an alternative to the Docker runtime, designed for server environments with the most rigorous security and production requirements.

**Applications**

* [Adobe Spindle](https://github.com/adobe-research/spindle) - Next-generation web analytics processing with Scala, Spark, and Parquet.
* [Apache Kiji](http://www.kiji.org/) - framework to collect and analyze data in real-time, based on HBase.
* [Apache Nutch](http://nutch.apache.org/) - open source web crawler.
* [Apache OODT](http://oodt.apache.org/) - capturing, processing and sharing of data for NASA's scientific archives.
* [Apache Tika](https://tika.apache.org/) - content analysis toolkit.
* [Domino](http://www.dominoup.com/) - Run, scale, share, and deploy models Ñ without any infrastructure..
* [Eclipse BIRT](http://www.eclipse.org/birt/) - Eclipse-based reporting system.
* [Eventhub](https://github.com/Codecademy/EventHub) - open source event analytics platform.
* [HIPI Library](http://hipi.cs.virginia.edu/) - API for performing image processing tasks on Hadoop's MapReduce.
* [Hunk](http://www.splunk.com/download/hunk) - Splunk analytics for Hadoop.
* [MADlib](http://madlib.net/community/) - data-processing library of an RDBMS to analyze data.
* [PivotalR](https://github.com/gopivotal/PivotalR) - R on Pivotal HD / HAWQ and PostgreSQL.
* [Qubole](http://www.qubole.com/) - auto-scaling Hadoop cluster, built-in data connectors.
* [Sense](https://senseplatform.com/) - Cloud Platform for Data Science and Big Data Analytics.
* [Snowplow](https://github.com/snowplow/snowplow) - enterprise-strength web and event analytics, powered by Hadoop, Kinesis, Redshift and Postgres.
* [SparkR](http://amplab-extras.github.io/SparkR-pkg/) - R frontend for Spark.
* [Splunk](http://www.splunk.com/) - analyzer for machine-generated date.
* [Talend](http://www.talend.com/products/big-data) - unified open source environment for YARN, Hadoop, HBASE, Hive, HCatalog & Pig.

**Search engine and framework**

* [Algolia](https://www.algolia.com/) - Hosted Search API that delivers instant and relevant results from the first keystroke.
* [Apache Blur](https://incubator.apache.org/blur/) - a search engine capable of querying massive amounts of structured data at incredible speeds.
* [Apache Lucene](http://lucene.apache.org/) - Search engine library.
* [Apache Solr](http://lucene.apache.org/solr/) - Search platform for Apache Lucene.
* [ElasticSearch](http://www.elasticsearch.org/) - Search and analytics engine based on Apache Lucene.
* [Elasticsearch Hadoop](https://github.com/elasticsearch/elasticsearch-hadoop) - Elasticsearch real-time search and analytics natively integrated with Hadoop. Supports Map/Reduce, Cascading, Apache Hive and Apache Pig..
* [Enigma.io](http://enigma.io/) - Freemium robust web application for exploring, filtering, analyzing, searching and exporting massive datasets scraped from across the Web.
* [Facebook Unicorn](https://www.facebook.com/publications/219621248185635/) - social graph search platform.
* [Google Caffeine](http://googleblog.blogspot.it/2010/06/our-new-search-index-caffeine.html) - continuous indexing system.
* [Google Percolator](http://research.google.com/pubs/pub36726.html) - continuous indexing system.
* [TeraGoogle](https://github.com/zenkay/bigdata-ecosystem/blob/master) - large search index.
* [Haeinsa](https://github.com/VCNC/haeinsa) - linearly scalable multi-row, multi-table transaction library for HBase based on Percolator.
* [HBase Coprocessor](https://blogs.apache.org/hbase/entry/coprocessor_introduction) - implementation of Percolator, part of HBase.
* [hIndex](https://github.com/Huawei-Hadoop/hindex) - Secondary Index for HBase.
* [SF1R Search Engine](http://github.com/izenecloud/sf1r-lite) - distributed search engine written in c++.
* [Lily HBase Indexer](http://ngdata.github.io/hbase-indexer/) - quickly and easily search for any content stored in HBase.
* [LinkedIn Bobo](http://senseidb.github.io/bobo/) - is a Faceted Search implementation written purely in Java, an extension to Apache Lucene.
* [LinkedIn Cleo](https://github.com/linkedin/cleo) - is a flexible software library for enabling rapid development of partial, out-of-order and real-time typeahead search.
* [LinkedIn Galene](http://engineering.linkedin.com/search/did-you-mean-galene) - search architecture at LinkedIn.
* [LinkedIn Zoie](https://github.com/senseidb/zoie) - is a realtime search/indexing system written in Java.
* [Sphinx Search Server](http://sphinxsearch.com/) - fulltext search engine.

**MySQL forks and evolutions**

* [Amazon Aurora](https://aws.amazon.com/rds/aurora/) - a MySQL-compatible, relational database engine that combines the speed and availability of high-end commercial databases with the simplicity and cost-effectiveness of open source databases.
* [Amazon RDS](http://aws.amazon.com/rds/) - MySQL databases in Amazon's cloud.
* [BigObject](http://www.bigobject.io/) - Real-time Computing Engine Designed for Big Data.
* [Drizzle](http://www.drizzle.org/) - evolution of MySQL 6.0.
* [Galera Cluster](http://galeracluster.com/) - a synchronous multi-master cluster for MySQL, Percona and MariaDB.
* [Google Cloud SQL](https://developers.google.com/cloud-sql/) - MySQL databases in Google's cloud.
* [HiveDB](http://www.hivedb.org/) - an open source framework for horizontally partitioning MySQL systems.
* [MariaDB](https://mariadb.org/) - enhanced, drop-in replacement for MySQL.
* [MySQL Cluster](http://www.mysql.com/products/cluster/) - MySQL implementation using NDB Cluster storage engine providing shared-nothing clustering and auto-sharding.
* [Percona Server](http://www.percona.com/software/percona-server) - enhanced, drop-in replacement for MySQL.
* [ProxySQL](https://github.com/renecannao/proxysql) - High Performance Proxy for MySQL.
* [TiDB](https://github.com/pingcap/tidb) - a distributed SQL database inspired by the design of Google F1.
* [TokuDB](http://www.tokutek.com/products/tokudb-for-mysql/) - TokuDB is a storage engine for MySQL and MariaDB.
* [WebScaleSQL](http://webscalesql.org/) - is a collaboration among engineers from several companies that face similar challenges in running MySQL at scale.
* [Youtube Vitess](https://github.com/youtube/vitess) - provides servers and tools which facilitate scaling of MySQL databases for large scale web services.

**PostgreSQL forks and evolutions**

* [HadoopDB](http://db.cs.yale.edu/hadoopdb/hadoopdb.html) - hybrid of MapReduce and DBMS.
* [IBM Netezza](http://www-01.ibm.com/software/data/netezza/) - high-performance data warehouse appliances.
* [Postgres-XL](http://www.postgres-xl.org/) - Scalable Open Source PostgreSQL-based Database Cluster.
* [RecDB](http://www-users.cs.umn.edu/~sarwat/RecDB/) - Open Source Recommendation Engine Built Entirely Inside PostgreSQL.
* [Stado](http://www.stormdb.com/community/stado) - open source MPP database system solely targeted at data warehousing and data mart applications.
* [Yahoo Everest](http://www.scribd.com/doc/3159239/70-Everest-PGCon-RT) - multi-peta-byte database / MPP derived by PostgreSQL.

**Memcached forks and evolutions**

* [Box Tron](http://engineering.opendns.com/2014/09/16/caching-scale/) - proxy to memcached servers.
* [Facebook McDipper](https://www.facebook.com/notes/facebook-engineering/mcdipper-a-key-value-cache-for-flash-storage/10151347090423920) - key/value cache for flash storage.
* [Facebook Mcrouter](https://github.com/facebook/mcrouter) - a memcached protocol router for scaling memcached deployments.
* [Facebook Memcached](https://www.facebook.com/notes/facebook-engineering/scaling-memcache-at-facebook/10151411410803920) - fork of Memcache.
* [Twemproxy](https://github.com/twitter/twemproxy) - A fast, light-weight proxy for memcached and redis.
* [Twitter Fatcache](https://github.com/twitter/fatcache) - key/value cache for flash storage.
* [Twitter Twemcache](https://github.com/twitter/twemcache) - fork of Memcache.

**Embedded Databases**

* [Actian PSQL](http://www.actian.com/products/operational-databases/) - ACID-compliant DBMS developed by Pervasive Software, optimized for embedding in applications.
* [BerkeleyDB](http://www.oracle.com/us/products/database/berkeley-db/overview/index.html) - a software library that provides a high-performance embedded database for key/value data.
* [eXtreme DB](http://www.mcobject.com/extremedbfamily.shtml) - in-memory database combines exceptional performance, reliability and developer efficiency in a proven real-time embedded database engine.
* [FairCom c-treeACE](http://www.faircom.com/ace/ace_t.php) - a cross-platform database engine.
* [Google Firebase](https://www.firebase.com/) - a powerful API to store and sync data in realtime.
* [HamsterDB](http://hamsterdb.com/) - transactional key-value database.
* [HanoiDB](https://github.com/krestenkrab/hanoidb) - Erlang LSM BTree Storage.
* [LevelDB](https://code.google.com/p/leveldb/) - a fast key-value storage library written at Google that provides an ordered mapping from string keys to string values.
* [LMDB](http://symas.com/mdb/) - ultra-fast, ultra-compact key-value embedded data store developed by Symas.
* [RocksDB](http://rocksdb.org/) - embeddable persistent key-value store for fast storage based on LevelDB.
* [TokioCabinet](http://fallabs.com/tokyocabinet/) - a library of routines for managing a database.
* [UnQLite](http://unqlite.org/) - a in-process software library which implements a self-contained, serverless, zero-configuration, transactional NoSQL database engine.

**Business Intelligence**

* [ActivePivot](http://quartetfs.com/products/activepivot) - Java In-Memory OLAP cube stored in columns, with clearly decoupled pre/post processing.
* [Adatao](http://adatao.com/pinsights.html) - business intelligence and data science platform.
* [Amazon QuickSight](https://aws.amazon.com/quicksight/) - Business Intelligence for Big Data.
* [Apama analytics](http://www.softwareag.com/corporate/products/bigdata/apama_analytics/overview/) - platform for streaming analytics and intelligent automated action.
* [Atigeo xPatterns](http://xpatterns.com/) - data analytics platform.
* [BIME Analytics](http://www.bimeanalytics.com/) - business intelligence platform in the cloud.
* [Chartio](https://chartio.com/) - lean business intelligence platform to visualize and explore your data.
* [Datapine](http://www.datapine.com/) - self-service business intelligence tool in the cloud.
* [Jaspersoft](https://www.jaspersoft.com/) - powerful business intelligence suite.
* [Jedox Palo](http://www.jedox.com/) - customisable Business Intelligence platform.
* [Lavastorm Analytics](http://www.lavastorm.com/) - used for audit analytics, revenue assurance, fraud management, and customer experience management.
* [LinkedIn GoSpeed](http://engineering.linkedin.com/performance/monitor-and-improve-web-performance-using-rum-data-visualization) - provides RUM data processing, visualization, monitoring, and analyses data daily, hourly, or on a near real-time basis.
* [Map-D](http://www.map-d.com/) - GPU in-memory database, big data analysis and visualization platform.
* [Microsoft](http://www.microsoft.com/en-us/server-cloud/solutions/business-intelligence/default.aspx) - business intelligence software and platform.
* [Microstrategy](http://www.microstrategy.com/) - software platforms for business intelligence, mobile intelligence, and network applications.
* [Pentaho](http://www.pentaho.com/) - business intelligence platform.
* [Qlik](http://www.qlik.com/) - business intelligence and analytics platform.
* [SpagoBI](http://www.spagoworld.org/xwiki/bin/view/SpagoBI/) - open source business intelligence platform.
* [Spotfire](http://spotfire.tibco.com/) - business intelligence platform.
* [Stratio Explorer](http://docs.stratio.com/modules/explorer/development) - an Interactive Web interpreter to Apache Crossdata, Stratio Ingestion, Stratio Decision,Markdown, Apache Spark, Apache Spark-SQL and command Shell.
* [Tableau](https://www.tableausoftware.com/) - business intelligence platform.
* [Teradata Aster](http://it.teradata.com/Teradata-Aster-Database/) - Big Data Analytics.
* [Tessera](http://tesseradata.org/) - Environment for Deep Analysis of Large Complex Data.
* [Zeppelin](http://zeppelin-project.org/) - open source data analysis environment on top of Hadoop..
* [Zoomdata](http://www.zoomdata.com/) - Big Data Analytics.

**Data Analysis**

* [Apache Zeppelin](http://zeppelin.incubator.apache.org/) - a web-based notebook that enables interactive data analytics.
* [Datameer](http://www.datameer.com/product/index.html) - data analytics application for Hadoop combines self-service data integration, analytics and visualization.
* [Ibis](http://www.ibis-project.org/) - Python big data analysis framework for high performance at Hadoop-scale, with first-class integration with Impala.
* [LinkedIn Pinot](http://engineering.linkedin.com/analytics/real-time-analytics-massive-scale-pinot) - a distributed system that supports columnar indexes with the ability to add new types of indexes.
* [Microsoft Cortana Analytics](http://www.microsoft.com/en-us/server-cloud/cortana-analytics-suite/overview.aspx) - a fully managed big data and advanced analytics suite that enables you to transform your data into intelligent action..
* [Myria](http://myria.cs.washington.edu/) - scalable Analytics-as-a-Service platform based on relational algebra.
* [Periscope](https://www.periscope.io/) - plugs directly into your databases and lets you run, save, and share analyses over billions of data rows in seconds.
* [Pinalytics](http://engineering.pinterest.com/post/104418761649/building-pinalytics-pinterests-data-analytics) - Pinterestâ��s data analytics engine.
* [Shiny](http://shiny.rstudio.com/) - web application framework for R.
* [Stratio Sparkta](http://docs.stratio.com/modules/sparkta/development) - real time monitoring.
* [Tamr](http://www.tamr.com/) - standalone tool to catalog all of your enterprise metadata.
* [Zaloni Bedrock](http://www.zaloni.com/product/bedrock) - fully integrated Hadoop data management platform.
* [Zaloni Mica](http://www.zaloni.com/product/mica) - self-service data discovery, curation, and governance.
* [Zillabyte](http://zillabyte.com/) - an API for distributed data computation. Scale with your data..

**Data Warehouse**

* [Google Mesa](http://static.googleusercontent.com/media/research.google.com/en/us/pubs/archive/42851.pdf) - highly scalable analytic data warehousing system.
* [IBM BigInsights](http://www.ibm.com/software/data/infosphere/biginsights/) - data processing, warehousing and analytics.
* [IBM dashDB](https://cloudant.com/dashdb/) - Data Warehousing and Analysis Needs, all in the Cloud.
* [Microsoft Azure SQL Data Warehouse](http://techcrunch.com/2015/04/29/microsoft-introduces-azure-sql-data-warehouse/) - businesses access to an elastic petabyte-scale, data warehouse-as-a-service offering that can scale according to their needs.
* [Microsoft Cosmos](http://research.microsoft.com/en-us/events/fs2011/helland_cosmos_big_data_and_big_challenges.pdf) - Microsoft's internal BigData analysis platform.

**Data Visualization**

* [Arbor](https://github.com/samizdatco/arbor) - graph visualization library using web workers and jQuery.
* [C3](http://c3js.org/) - D3-based reusable chart library.
* [CartoDB](https://github.com/CartoDB/cartodb) - open-source or freemium hosting for geospatial databases with powerful front-end editing capabilities and a robust API.
* [Chart.js](http://www.chartjs.org/) - open source HTML5 Charts visualizations.
* [Chartist.js](https://github.com/gionkunz/chartist-js) - another open source HTML5 Charts visualization.
* [Crossfilter](http://square.github.io/crossfilter/) - avaScript library for exploring large multivariate datasets in the browser. Works well with dc.js and d3.js.
* [Cubism](https://github.com/square/cubism) - JavaScript library for time series visualization.
* [Cytoscape](http://cytoscape.github.io/) - JavaScript library for visualizing complex networks.
* [D3](http://d3js.org/) - javaScript library for manipulating documents.
* [DC.js](http://dc-js.github.io/dc.js/) - Dimensional charting built to work natively with crossfilter rendered using d3.js. Excellent for connecting charts/additional metadata to hover events in D3.
* [Envisionjs](https://github.com/HumbleSoftware/envisionjs) - dynamic HTML5 visualization.
* [FnordMetric ChartSQL](http://fnordmetric.io/chartsql/) - allows you to write SQL queries that return charts instead of tables. The charts are rendered as SVG vector graphics..
* [Freeboard](https://github.com/Freeboard/freeboard) - open source real-time dashboard builder for IOT and other web mashups.
* [Gephi](https://github.com/gephi/gephi) - An award-winning open-source platform for visualizing and manipulating large graphs and network connections.
* [Google Charts](https://developers.google.com/chart/) - simple charting API.
* [Grafana](http://grafana.org/) - open source, feature rich metrics dashboard and graph editor for Graphite, InfluxDB & OpenTSDB.
* [Graphistry](http://www.graphistry.com/) - running on GPUs and turns static designs into interactive tools using client/cloud GPU infrastructure and GPU-accelerated languages like Superconductor.
* [Graphite](http://graphite.wikidot.com/) - scalable Realtime Graphing.
* [Highcharts](http://www.highcharts.com/) - simple and flexible charting API.
* [IPython](http://ipython.org/) - provides a rich architecture for interactive computing.
* [Keylines](http://keylines.com/) - toolkit for visualizing the networks in your data.
* [Kibana](http://www.elasticsearch.org/overview/kibana/) - visualize logs and time-stamped data.
* [Matplotlib](https://github.com/matplotlib/matplotlib) - plotting with Python.
* [Microsoft SandDance](https://www.sanddance.ms/) - visually explore data sets to find stories and extract insights.
* [NVD3](http://nvd3.org/) - chart components for d3.js.
* [Peity](https://github.com/benpickles/peity) - Progressive SVG bar, line and pie charts.
* [Plot.ly](http://plot.ly/) - Easy-to-use web service that allows for rapid creation of complex charts, from heatmaps to histograms. Upload data to create and style charts with Plotly's online spreadsheet. Fork others' plots..
* [Recline](https://github.com/okfn/recline) - simple but powerful library for building data applications in pure Javascript and HTML.
* [Redash](https://github.com/everythingme/redash) - open-source platform to query and visualize data.
* [Sigma.js](https://github.com/jacomyal/sigma.js) - JavaScript library dedicated to graph drawing.
* [Square Cubism.js](http://square.github.io/cubism/) - aÊD3Êplugin for visualizing time series. Use Cubism to construct better realtime dashboards, pulling data fromÊGraphite,ÊCubeÊand other sources.
* [Stratio Viewer](http://www.stratio.com/viewer/kbase/) - dashboarding tool.
* [Vega](https://github.com/trifacta/vega) - a visualization grammar.

**Internet of Things**

* [2lemetry](http://2lemetry.com/) - Platform for Internet of things.
* [Evrything](https://evrythng.com/) - Making products smart.
* [ThingWorx](http://www.thingworx.com/) - Rapid development and connection of intelligent systems.

**Papers**

* [Published in 2016](https://github.com/zenkay/bigdata-ecosystem#2016)
* [Published in 2015](https://github.com/zenkay/bigdata-ecosystem#2015)
* [Published in 2014](https://github.com/zenkay/bigdata-ecosystem#2014)
* [Published in 2013](https://github.com/zenkay/bigdata-ecosystem#2013)
* [Published in 2012](https://github.com/zenkay/bigdata-ecosystem#2012)
* [Published in 2011](https://github.com/zenkay/bigdata-ecosystem#2011)
* [Published in 2010](https://github.com/zenkay/bigdata-ecosystem#2010)
* [Published in 2009](https://github.com/zenkay/bigdata-ecosystem#2009)
* [Published in 2008](https://github.com/zenkay/bigdata-ecosystem#2008)
* [Published in 2007](https://github.com/zenkay/bigdata-ecosystem#2007)
* [Published in 2006](https://github.com/zenkay/bigdata-ecosystem#2006)
* [Published in 2005](https://github.com/zenkay/bigdata-ecosystem#2005)
* [Published in 2004](https://github.com/zenkay/bigdata-ecosystem#2004)
* [Published in 2003](https://github.com/zenkay/bigdata-ecosystem#2003)
* [Published in 2002](https://github.com/zenkay/bigdata-ecosystem#2002)
* [Published in 2001](https://github.com/zenkay/bigdata-ecosystem#2001)
* [Published in 2000](https://github.com/zenkay/bigdata-ecosystem#2000)
* [Published in 1999](https://github.com/zenkay/bigdata-ecosystem#1999)
* [Published in 1998](https://github.com/zenkay/bigdata-ecosystem#1998)
* [Published in 1997](https://github.com/zenkay/bigdata-ecosystem#1997)

**2016**

* **2016** - [Learning to Simplify: Fully Convolutional Networks for Rough Sketch Cleanup](http://hi.cs.waseda.ac.jp/~esimo/publications/SimoSerraSIGGRAPH2016.pdf)
* **2016** - [Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples](http://arxiv.org/pdf/1602.02697v2.pdf)
* **2016** - [Understanding Deep Convolutional Networks](http://arxiv.org/pdf/1601.04920v1.pdf)

**2015**

* **2015** - [A Neural Algorithm of Artistic Style](http://arxiv.org/pdf/1508.06576v1.pdf)
* **2015** - [Deep Image: Scaling up Image Recognition](http://arxiv.org/pdf/1501.02876v1.pdf)
* **2015** - [Deep Speech 2: End-to-End Speech Recognition in English and Mandarin](http://arxiv.org/pdf/1512.02595v1.pdf)
* **2015** - [Deep Speech: Scaling up end-to-end speech recognition](http://arxiv.org/pdf/1412.5567.pdf)
* **2015** - [Fast Convolutional Nets With fbfft: A GPU Performance Evaluation](http://arxiv.org/pdf/1412.7580v2.pdf)
* **2015** - [G-OLA: Generalized On-Line Aggregation for Interactive Analysis on Big Data](https://amplab.cs.berkeley.edu/wp-content/uploads/2015/05/sigmod_2015.pdf)
* **2015** - [Giraffe: Using Deep Reinforcement Learning to Play Chess](http://arxiv.org/pdf/1509.01549v2.pdf)
* **2015** - [Hidden Technical Debt in Machine Learning Systems](http://papers.nips.cc/paper/5656-hidden-technical-debt-in-machine-learning-systems.pdf)
* **2015** - [Klout Score: Measuring Influence Across Multiple Social Networks](http://arxiv.org/pdf/1510.08487v1.pdf)
* **2015** - [Large-scale cluster management at Google with Borg](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/43438.pdf)
* **2015** - [Machine Learning Classification over Encrypted Data](http://www.internetsociety.org/sites/default/files/04_1_2.pdf)
* **2015** - [Machine Learning Methods for Computer Security](http://drops.dagstuhl.de/opus/volltexte/2013/3790/pdf/dagrep_v002_i009_p109_s12371.pdf)
* **2015** - [Neural Networks with Few Multiplications](http://arxiv.org/pdf/1510.03009v1.pdf)
* **2015** - [Self-Repairing Disk Arrays](http://arxiv.org/pdf/1501.00513v1.pdf)
* **2015** - [Spark SQL: Relational Data Processing in Spark](http://people.csail.mit.edu/matei/papers/2015/sigmod_spark_sql.pdf)
* **2015** - [SparkNetwork: Training Deep Network in Spark](http://arxiv.org/pdf/1511.06051v2.pdf)
* **2015** - [Succinct: Enabling Queries on Compressed Data](https://www.usenix.org/system/files/conference/nsdi15/nsdi15-paper-agarwal.pdf)
* **2015** - [Taming the Wild: A Unified Analysis of HOGWILD!-Style Algorithms](http://arxiv.org/pdf/1506.06438v1.pdf%20)
* **2015** - [The Missing Piece in Complex Analytics: Low Latency, Scalable Model Management and Serving with Velox](http://arxiv.org/pdf/1409.3809v2.pdf)
* **2015** - [Trill: A High-Performance Incremental Query Processor for Diverse Analytics](http://research.microsoft.com/pubs/231690/trill-vldb2015.pdf)
* **2015** - [Twitter Heron: Stream Processing at Scale](http://delivery.acm.org/10.1145/2750000/2742788/p239-kulkarni.pdf)

**2014**

* **2014** - [3D Object Manipulation in a Single Photograph using Stock 3D Models](https://www.cs.cmu.edu/~om3d/papers/SIGGRAPH2014.pdf)
* **2014** - [A Partitioning Framework for Aggressive Data Skipping](http://www.vldb.org/pvldb/vol7/p1617-sun.pdf?imm_mid=0c5589&cmp=em-strata-na-na-newsltr_20141022)
* **2014** - [A Sample-and-Clean Framework for Fast and Accurate Query Processing on Dirty Data](https://amplab.cs.berkeley.edu/wp-content/uploads/2014/05/sampleclean-sigmod14.pdf)
* **2014** - [A Self-Configurable Geo-Replicated Cloud Storage System](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-ardekani.pdf)
* **2014** - [All File Systems Are Not Created Equal: On the Complexity of Crafting Crash-Consistent Applications](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-pillai.pdf)
* **2014** - [Arrakis: The Operating System is the Control Plane](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-peter_simon.pdf)
* **2014** - [Automatic Construction of Inference-Supporting Knowledge Bases](http://allenai.org/content/publications/clark_balasubramanian.pdf)
* **2014** - [Bayesian group latent factor analysis with structured sparse priors](http://arxiv.org/pdf/1411.2698v1.pdf)
* **2014** - [Chinese Open Relation Extraction for Knowledge Acquisition](http://allenai.org/content/publications/10094_Paper.pdf)
* **2014** - [Coordination Avoidance in Database Systems](http://www.bailis.org/papers/ca-vldb2015.pdf)
* **2014** - [DeepFace: Closing the Gap to Human-Level Performance in Face Verification](http://www.cs.toronto.edu/~ranzato/publications/taigman_cvpr14.pdf)
* **2014** - [Diagram Understanding in Geometry Questions](http://allenai.org/content/publications/diagram_understanding_in_geometry_questions.pdf)
* **2014** - [Discourse Complements Lexical Semantics for Non-factoid Answer Reranking](http://allenai.org/content/publications/acl2014.pdf)
* **2014** - [Do we Need Hundreds of Classifiers to Solve Real World Classification Problems?](http://jmlr.org/papers/volume15/delgado14a/delgado14a.pdf)
* **2014** - [Eidetic Systems](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-devecsery.pdf)
* **2014** - [Execution Primitives for Scalable Joins and Aggregations in Map Reduce](http://www.vldb.org/pvldb/vol7/p1462-vemuri.pdf)
* **2014** - [Extracting More Concurrency from Distributed Transactions](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-mu.pdf)
* **2014** - [f4: Facebookâ��s Warm BLOB Storage System](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-muralidhar.pdf)
* **2014** - [Fast Databases with Fast Durability and Recovery Through Multicore Parallelism](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-zheng_wenting.pdf)
* **2014** - [Fastpass: A Centralized "Zero-Queue" Datacenter Network](http://fastpass.mit.edu/Fastpass-SIGCOMM14-Perry.pdf)
* **2014** - [First-person Hyper-lapse Videos](http://research.microsoft.com/en-us/um/redmond/projects/hyperlapse/paper/hyperlapse.pdf)
* **2014** - [GloVe: Global Vectors for Word Representation](http://nlp.stanford.edu/projects/glove/glove.pdf)
* **2014** - [GraphX: Graph Processing in a Distributed Dataflow Framework](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-gonzalez.pdf)
* **2014** - [Guess Who Rated This Movie: Identifying Users Through Subspace Clustering](http://arxiv.org/pdf/1408.2055v1.pdf)
* **2014** - [In Search of an Understandable Consensus Algorithm](https://www.usenix.org/system/files/conference/atc14/atc14-paper-ongaro.pdf)
* **2014** - [Learning Everything about Anything: Webly-Supervised Visual Concept Learning](http://allenai.org/content/publications/objectNgrams_cvpr14.pdf)
* **2014** - [Learning to Solve Arithmetic Word Problems with Verb Categorization](http://allenai.org/content/publications/Arithmetic_Word_Problems.pdf)
* **2014** - [Log-structured Memory for DRAM-based Storage](https://www.usenix.org/system/files/conference/fast14/fast14-paper_rumble.pdf)
* **2014** - [Logical Physical Clocks and Consistent Snapshots in Globally Distributed Databases](http://www.cse.buffalo.edu/tech-reports/2014-04.pdf)
* **2014** - [MapGraph: A High Level API for Fast Development of High Performance Graph Analytics on GPUs](http://mapgraph.io/papers/MapGraph-SIGMOD-2014.pdf)
* **2014** - [Mesa: Geo-Replicated, Near Real-Time, Scalable Data Warehousing](https://static.googleusercontent.com/media/research.google.com/en/pubs/archive/42851.pdf)
* **2014** - [Modeling Biological Processes for Reading Comprehension](http://allenai.org/content/publications/berant-srikumar-manning-emnlp14.pdf)
* **2014** - [Orca A Modular Query Optimizer Architecture for Big Data](http://www.pivotal.io/sites/default/files/SIGMODMay2014HAWQAdvantages.pdf)
* **2014** - [Pigeon: A Spatial MapReduce Language](http://spatialhadoop.cs.umn.edu/publications/ICDE14_demo_763.pdf)
* **2014** - [Project Adam: Building an Efficient and Scalable Deep Learning Training System](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-chilimbi.pdf)
* **2014** - [Quantum Deep Learning](http://arxiv.org/pdf/1412.3489v1.pdf)
* **2014** - [R Markdown: Integrating A Reproducible Analysis Tool into Introductory Statistics](http://arxiv.org/pdf/1402.1894v1.pdf)
* **2014** - [Salt: Combining ACID and BASE in a Distributed Database](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-xie.pdf)
* **2014** - [Scalable Object Detection using Deep Neural Networks](http://www.cv-foundation.org/openaccess/content_cvpr_2014/papers/Erhan_Scalable_Object_Detection_2014_CVPR_paper.pdf)
* **2014** - [Sequence to Sequence Learning with Neural Networks](http://arxiv.org/pdf/1409.3215v1.pdf)
* **2014** - [Show and Tell: A Neural Image Caption Generator](http://arxiv.org/pdf/1411.4555v1.pdf)
* **2014** - [Simple Testing Can Prevent Most Critical Failures: An Analysis of Production Failures in Distributed Data-Intensive Systems](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-yuan.pdf)
* **2014** - [The Mystery Machine: End-to-end Performance Analysis of Large-scale Internet Services](https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-chow.pdf)
* **2014** - [The Trill Incremental Analytics Engine](http://research.microsoft.com/pubs/214609/trill-TR.pdf)

**2013**

* **2013** - [A Demonstration of SpatailHadoop: An Efficient MapReduce Framework for Spatial Data](http://spatialhadoop.cs.umn.edu/publications/p744-eldawy.pdf)
* **2013** - [A Lightweight and High Performance Monolingual Word Aligner](http://www.cs.utexas.edu/users/pclark/papers/yao-jacana-wordalign-acl2013.pdf)
* **2013** - [Answer Extraction as Sequence Tagging with Tree Edit Distance](http://www.cs.utexas.edu/users/pclark/papers/yao-jacana-qa-naacl2013.pdf)
* **2013** - [Automatic Coupling of Answer Extraction and Information Retrieval](http://www.cs.utexas.edu/users/pclark/papers/yao-jacana-ir-acl2013.pdf)
* **2013** - [CG\_Hadoop: Computational Geometry in MapReduce](http://spatialhadoop.cs.umn.edu/publications/p0144_Eldawy.pdf)
* **2013** - [Consistency-Based Service Level Agreements for Cloud Storage](http://delivery.acm.org/10.1145/2530000/2522731/p309-terry.pdf)
* **2013** - [Dimension Independent Matrix Square using MapReduce](http://arxiv.org/pdf/1304.1467v2.pdf)
* **2013** - [Druid A Real-time Analytical Data Store](http://static.druid.io/docs/druid.pdf)
* **2013** - [Efficient Estimation of Word Representations in Vector Space](http://arxiv.org/pdf/1301.3781.pdf)
* **2013** - [Event labeling combining ensemble detectors and background knowledge](http://link.springer.com/content/pdf/10.1007%2Fs13748-013-0040-3.pdf)
* **2013** - [Everything You Always Wanted to Know About Synchronization but Were Afraid to Ask](http://sigops.org/sosp/sosp13/papers/p33-david.pdf)
* **2013** - [F1: A Distributed SQL Database That Scales](http://stevereads.com/papers_to_read/f1_a_distributed_sql_database_that_scales.pdf)
* **2013** - [Fast Training of Convolutional Networks through FFTs](http://arxiv.org/pdf/1312.5851v5.pdf)
* **2013** - [GraphX: A Resilient Distributed Graph System on Spark](https://amplab.cs.berkeley.edu/wp-content/uploads/2013/05/grades-graphx_with_fonts.pdf)
* **2013** - [HyperLogLog in Practice: Algorithmic Engineering of a State of The Art Cardinality 2013 Estimation Algorithm](http://stefanheule.com/papers/edbt2013-hyperloglog.pdf)
* **2013** - [MillWheel: Fault-Tolerant Stream Processing at Internet Scale](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/41378.pdf)
* **2013** - [MLbase: A Distributed Machine-learning System](http://cidrdb.org/cidr2013/Papers/CIDR13_Paper118.pdf)
* **2013** - [Naiad: A Timely Dataflow System](http://research.microsoft.com/pubs/201100/naiad_sosp2013.pdf)
* **2013** - [Omega: flexible, scalable schedulers for large compute clusters](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/41684.pdf)
* **2013** - [Online, Asynchronous Schema Change in F1](http://db.disi.unitn.eu/pages/VLDBProgram/pdf/industry/p764-rae.pdf)
* **2013** - [Presto: Distributed Machine Learning and Graph Processing with Sparse Matrices](http://eurosys2013.tudos.org/wp-content/uploads/2013/paper/Venkataraman.pdf)
* **2013** - [Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank](http://nlp.stanford.edu/~socherr/EMNLP2013_RNTN.pdf)
* **2013** - [Rich feature hierarchies for accurate object detection and semantic segmentation](http://arxiv.org/pdf/1311.2524v4.pdf)
* **2013** - [Scalable Progressive Analytics on Big Data in the Cloud](http://research.microsoft.com/pubs/200169/now-vldb.pdf)
* **2013** - [Scaling Memcache at Facebook](https://www.usenix.org/system/files/conference/nsdi13/nsdi13-final170_update.pdf&sa=U&ei=gWJjU97pOeqxsQSDkYDAAg&ved=0CBsQFjAA&usg=AFQjCNGMeuWne9ywncbgux_XiZW6lQWHNw)
* **2013** - [Scuba: Diving into Data at Facebook](http://db.disi.unitn.eu/pages/VLDBProgram/pdf/industry/p767-wiener.pdf)
* **2013** - [Semi-Markov Phrase-based Monolingual Alignment](http://www.cs.utexas.edu/users/pclark/papers/EMNLP2013-phraseAlign.pdf)
* **2013** - [Shark: SQL and Rich Analytics at Scale](http://www.eecs.berkeley.edu/Pubs/TechRpts/2012/EECS-2012-214.pdf)
* **2013** - [Some Improvements on Deep Convolutional Neural Network Based Image Classification](http://arxiv.org/pdf/1312.5402v1.pdf)
* **2013** - [Sparrow: Distributed, Low Latency Scheduling](http://delivery.acm.org/10.1145/2530000/2522716/p69-ousterhout.pdf)
* **2013** - [Sparrow: Scalable Scheduling for Sub-Second Parallel Jobs](http://www.eecs.berkeley.edu/Pubs/TechRpts/2013/EECS-2013-29.pdf)
* **2013** - [TAO: Facebookâ��s Distributed Data Store for the Social Graph](https://www.cs.cmu.edu/~pavlo/courses/fall2013/static/papers/11730-atc13-bronson.pdf)
* **2013** - [Toward Common Patterns for Distributed, Concurrent, Fault-Tolerant Code](http://www.scs.stanford.edu/~stutsman/papers/stutsman-dcft-hotos13.pdf)
* **2013** - [Unicorn: A System for Searching the Social Graph](http://db.disi.unitn.eu/pages/VLDBProgram/pdf/industry/p871-curtiss.pdf)
* **2013** - [Warp: Lightweight Multi-Key Transactions for Key-Value Stores](http://hyperdex.org/papers/warp.pdf)

**2012**

* **2012** - [A Few Useful Things to Know about Machine Learning](http://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf)
* **2012** - [A Sublinear Time Algorithm for PageRank Computations](http://research.microsoft.com/en-us/um/people/borgs/Papers/SublinearPR.pdf)
* **2012** - [Avatara: OLAP for Web-scale Analytics Products](http://www.vldb.org/pvldb/vol5/p1874_liliwu_vldb2012.pdf)
* **2012** - [Blink and It's Done. Interactive Queries on Very Large Data](http://www.cs.berkeley.edu/~sameerag/blinkdb_vldb12_demo.pdf)
* **2012** - [BlinkDB: Queries with Bounded Errors and Bounded Response Times on Very Large Data](https://www.cs.berkeley.edu/~sameerag/blinkdb_eurosys13.pdf)
* **2012** - [Building high-level features using large scale unsupervised learning](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/38115.pdf)
* **2012** - [Dimension Independent Similarity Computation](http://arxiv.org/pdf/1206.2082.pdf)
* **2012** - [Earlybird: Real-Time Search at Twitter](http://www.umiacs.umd.edu/~jimmylin/publications/Busch_etal_ICDE2012.pdf)
* **2012** - [Fast and Interactive Analytics over Hadoop Data with Spark](https://www.usenix.org/system/files/login/articles/zaharia.pdf)
* **2012** - [HyperDex: A Distributed, Searchable Key-Value Store](http://hyperdex.org/papers/hyperdex.pdf)
* **2012** - [ImageNet Classification with Deep Convolutional Neural Networks](http://www.cs.toronto.edu/~fritz/absps/imagenet.pdf)
* **2012** - [Large Scale Distributed Deep Networks](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/40565.pdf)
* **2012** - [Large:Scale Machine Learning at Twitter](http://www.umiacs.umd.edu/~jimmylin/publications/Lin_Kolcz_SIGMOD2012.pdf)
* **2012** - [Multi-Scale Matrix Sampling and Sublinear-Time PageRank Computation](http://arxiv.org/pdf/1202.2771v5.pdf)
* **2012** - [Paxos Made Parallel](http://research.microsoft.com/pubs/178045/ppaoxs-paper29.pdf)
* **2012** - [Paxos Replicated State Machines as the Basis of a High-Performance Data Store](https://www.usenix.org/legacy/events/nsdi11/tech/full_papers/Bolosky.pdf)
* **2012** - [Perspectives on the CAP Theorem](http://groups.csail.mit.edu/tds/papers/Gilbert/Brewer2.pdf)
* **2012** - [Processing a Trillion Cells per Mouse Click](http://vldb.org/pvldb/vol5/p1436_alexanderhall_vldb2012.pdf)
* **2012** - [Shark: Fast Data Analysis Using Coarse-grained Distributed Memory](http://www.cs.berkeley.edu/~matei/papers/2012/sigmod_shark_demo.pdf)
* **2012** - [Spanner: Google's Globally-Distributed Database](http://static.googleusercontent.com/media/research.google.com/en/archive/spanner-osdi2012.pdf)
* **2012** - [Temporal Analytics on Big Data for Web Advertising](http://research.microsoft.com/pubs/155806/timr-icde2012.pdf)
* **2012** - [The Unified Logging Infrastructure for Data Analytics at Twitter](http://vldb.org/pvldb/vol5/p1771_georgelee_vldb2012.pdf)
* **2012** - [The Vertica Analytic Database- C-Store 7 Years Later](http://vldb.org/pvldb/vol5/p1790_andrewlamb_vldb2012.pdf)

**2011**

* **2011** - [Consistency, Availability, and Convergence](http://www.cs.utexas.edu/users/dahlin/papers/cac-tr.pdf)
* **2011** - [CrowdDB: Answering Queries with Crowdsourcing](http://csce.uark.edu/~tingxiny/courses/5013spring13/readingList/crowddb_sigmod2011.pdf)
* **2011** - [CrowdDB: Query Processing with the VLDB Crowd](http://cs.brown.edu/~kraskat/pub/vldb11-crowddb_demo.pdf)
* **2011** - [Fast Crash Recovery in RAMCloud](http://web.stanford.edu/~ouster/cgi-bin/papers/ramcloud-recovery.pdf)
* **2011** - [Hogwild!: A Lock-Free Approach to Parallelizing Stochastic Gradient Descent](http://www.eecs.berkeley.edu/~brecht/papers/hogwildTR.pdf)
* **2011** - [It's Time for Low Latency](http://www.scs.stanford.edu/~rumble/papers/latency_hotos11.pdf)
* **2011** - [Matching Unstructured Product Offers to Structured Product Specifications](http://research.microsoft.com/pubs/144534/matching_tr.pdf)
* **2011** - [Megastore: Providing Scalable, Highly Available Storage for Interactive Services](http://www.cidrdb.org/cidr2011/Papers/CIDR11_Paper32.pdf)
* **2011** - [Resilient Distributed Datasets- A Fault-Tolerant Abstraction for In-Memory Cluster Computing](https://www.cs.berkeley.edu/~matei/papers/2012/nsdi_spark.pdf)
* **2011** - [Scarlett: Coping with Skewed Content Popularity in MapReduce Clusters](http://research.microsoft.com/en-us/um/people/srikanth/data/scarlett_eurosys11.pdf)

**2010**

* **2010** - [A Method of Automated Nonparametric Content Analysis for Social Science](http://dash.harvard.edu/bitstream/handle/1/5125261/method%20.pdf?sequence=1)
* **2010** - [Dapper, a Large-Scale Distributed Systems Tracing Infrastructure](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/36356.pdf)
* **2010** - [Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers](http://web.stanford.edu/~boyd/papers/pdf/admm_distr_stats.pdf)
* **2010** - [Dremel: Interactive Analysis of Web-Scale Datasets](http://static.googleusercontent.com/media/research.google.com/en/pubs/archive/36632.pdf)
* **2010** - [Finding a needle in Haystack- Facebook's photo storage](https://www.usenix.org/legacy/event/osdi10/tech/full_papers/Beaver.pdf)
* **2010** - [FlumeJava: Easy, Eff¥cient Data-Parallel Pipelines](http://pages.cs.wisc.edu/~akella/CS838/F12/838-CloudPapers/FlumeJava.pdf)
* **2010** - [Large:scale Incremental Processing Using Distributed Transactions and Notifications](https://www.usenix.org/legacy/event/osdi10/tech/full_papers/Peng.pdf)
* **2010** - [Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center](http://static.usenix.org/event/nsdi11/tech/full_papers/Hindman_new.pdf)
* **2010** - [Pregel: A System for Large-Scale Graph Processing](http://kowshik.github.io/JPregel/pregel_paper.pdf)
* **2010** - [S4: Distributed Stream Computing Platform](http://www.4lunas.org/pub/2010-s4.pdf)
* **2010** - [Spark: Cluster Computing with Working Sets](http://www.cs.berkeley.edu/~matei/papers/2010/hotcloud_spark.pdf)
* **2010** - [The Learning Behind Gmail Priority Inbox](http://static.googleusercontent.com/media/research.google.com/fr/pubs/archive/36955.pdf)
* **2010** - [ZooKeeper: Wait-free coordination for Internet-scale systems](https://www.usenix.org/legacy/event/usenix10/tech/full_papers/Hunt.pdf)

**2009**

* **2009** - [Cassandra - A Decentralized Structured Storage System](https://www.cs.cornell.edu/projects/ladis2009/papers/lakshman-ladis2009.pdf)
* **2009** - [Feature Hashing for Large Scale Multitask Learning](http://alex.smola.org/papers/2009/Weinbergeretal09.pdf)
* **2009** - [HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads](http://www.vldb.org/pvldb/2/vldb09-861.pdf)
* **2009** - [Vertical Paxos and Primary-Backup Replication](http://research.microsoft.com/en-us/um/people/lamport/pubs/vertical-paxos.pdf)

**2008**

* **2008** - [Chukwa: A large-scale monitoring system](http://mmm.csd.uwo.ca/courses/CS9842/papers/Paper-13-Ariel-Rabkin.pdf)
* **2008** - [Column:Stores vs. Row-Stores- How Different Are They Really?](http://db.csail.mit.edu/projects/cstore/abadi-sigmod08.pdf)
* **2008** - [PNUTS: Yahoo!Õs Hosted Data Serving Platform](http://www.mpi-sws.org/~druschel/courses/ds/papers/cooper-pnuts.pdf)
* **2008** - [Top 10 algorithms in data mining](http://www.cs.umd.edu/~samir/498/10Algorithms-08.pdf)

**2007**

* **2007** - [Architecture of a Database System](http://db.cs.berkeley.edu/papers/fntdb07-architecture.pdf)
* **2007** - [Consistent Streaming Through Time: A Vision for Event Stream Processing](http://research.microsoft.com/pubs/156569/cidr07p42.pdf)
* **2007** - [Dryad: Distributed Data-Parallel Programs from Sequential Building Blocks](http://cs.brown.edu/~debrabant/cis570-website/papers/dryad.pdf)
* **2007** - [Dynamo: Amazon's Highly Available Key-value Store](http://www.allthingsdistributed.com/files/amazon-dynamo-sosp2007.pdf)
* **2007** - [Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments](http://vis-www.cs.umass.edu/lfw/lfw.pdf)
* **2007** - [Life beyond Distributed Transactions: an ApostateÕs Opinion](http://www.ics.uci.edu/~cs223/papers/cidr07p15.pdf)
* **2007** - [Paxos Made Live - An Engineering Perspective](http://www.cs.utexas.edu/users/lorenzo/corsi/cs380d/papers/paper2-1.pdf)

**2006**

* **2006** - [Bigtable: A Distributed Storage System for Structured Data](http://static.googleusercontent.com/media/research.google.com/en/archive/bigtable-osdi06.pdf)
* **2006** - [Ceph: A Scalable, High-Performance Distributed File System](http://www.ssrc.ucsc.edu/Papers/weil-osdi06.pdf)
* **2006** - [Map-Reduce for Machine Learning on Multicore](http://machinelearning.wustl.edu/mlpapers/paper_files/NIPS2006_725.pdf)
* **2006** - [The Chubby lock service for loosely-coupled distributed systems](http://static.googleusercontent.com/media/research.google.com/en/archive/chubby-osdi06.pdf)

**2005**

* **2005** - [Fast Paxos](http://research.microsoft.com/pubs/64624/tr-2005-112.pdf)

**2004**

* **2004** - [Cheap Paxos](http://research.microsoft.com/en-us/um/people/lamport/pubs/web-dsn-submission.pdf)
* **2004** - [MapReduce: Simplified Data Processing on Large Clusters](http://static.googleusercontent.com/media/research.google.com/en/archive/mapreduce-osdi04.pdf)

**2003**

* **2003** - [Interpreting the Data: Parallel Analysis with Sawzall](http://static.googleusercontent.com/media/research.google.com/en/archive/sawzall-sciprog.pdf)
* **2003** - [The Google File System](http://static.googleusercontent.com/media/research.google.com/en/archive/gfs-sosp2003.pdf)

**2002**

* **2002** - [Brewer's Conjecture and the Feasibility of Consistent, Available, Partition-Tolerant Web Services](http://lpd.epfl.ch/sgilbert/pubs/BrewersConjecture-SigAct.pdf)

**2001**

* **2001** - [Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications](http://pdos.csail.mit.edu/papers/chord:sigcomm01/chord_sigcomm.pdf)
* **2001** - [Paxos Made Simple](http://research.microsoft.com/en-us/um/people/lamport/pubs/paxos-simple.pdf)
* **2001** - [Random Forrest](http://oz.berkeley.edu/~breiman/randomforest2001.pdf)

**1999**

* **1999** - [Pasting Small Votes for Classification in Large Databases and On-Line](http://link.springer.com/content/pdf/10.1023%2FA%3A1007563306331.pdf)
* **1999** - [The PageRank Citation Ranking: Bringing Order to the Web](http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf)

**1997**

* **1997** - [Application-Controlled Demand Paging for Out-of-Core Visualization](http://www.nas.nasa.gov/assets/pdf/techreports/1997/nas-97-010.pdf)