

**NoSQL Architecture Comparison**

**Course Number:** NSQL-110WA  
**Duration:** 2 days

**Overview**

The variety of NoSQL (Not Only SQL) technologies can be overwhelming. Which NoSQL platform should you choose? This NoSQL Architecture Comparison training class cuts through the hype to explain the architectures of NoSQL systems such as Pig, Hive, HBase, Cassandra, and MongoDB. Attendees learn how to make informed big data decisions and identify suitable NoSQL database use cases. By the end of this course, students are equipped to confidently select NoSQL persistence systems for their organization's needs.

**Prerequisites**

All attendees must have a background in enterprise information systems design.

**Materials**

All NoSQL Architecture training students will receive comprehensive courseware.

**Software Needed on Each Student PC**

* Computer with Internet connectivity
* Ability to install software on the computer
* Recent 64-bit OS, such as Windows 10, macOS, or Linux

**Objectives**

* Understand the core concepts of big data
* Explore the most common NoSQL stores
* Choose the correct NoSQL database for specific use cases
* Understand the architecture of Hadoop and MongoDB

**Outline**

* Introduction to NoSQL Systems
  + Gartner's Definition of Big Data
  + The V3
  + Properties
  + Limitations of Relational Databases
  + What are NoSQL Databases?
  + The Past and Present of the NoSQL World
  + NoSQL Database Properties
  + NoSQL Benefits
  + Use Cases for NoSQL Database Systems
  + NoSQL Database Storage Types
  + The CAP Theorem
  + Mechanisms to Guarantee a Single CAP Property
  + NoSQL Systems CAP Triangle
  + Limitations of NoSQL Databases
  + Mix-and-Match Approach
  + Big Data Sharding
  + Sharding Example
  + Google BigTable
  + BigTable-based Applications
  + BigTable Design
  + Barriers to Adoption
  + Dismantling Barriers to Adoption
  + Industry trends
  + NoSQL Technology Adoption Action Plan
* Apache HBase
  + What is HBase?
  + HBase Design
  + HBase Master (HMaster)
  + Sparse Data Sets
  + Regions and Region Servers
  + HBase Features
  + HBase High Availability
  + The Write-Ahead Log (WAL) and MemStore
  + HBase vs RDBS
  + Interfacing with HBase
  + HBase Thrift and REST Gateway
  + HBase Table Design
  + Column Families
  + A Cell's Value Versioning
  + Timestamps
  + Accessing Cells
  + HBase Table Design Digest
  + The Conceptual View of an HBase Table
  + HBase Compaction
  + Loading Data in HBase
  + Column Families Notes
  + Cardinality of Column Families
  + Hotspotting
  + Rowkey Design Notes
  + Security
  + HBase Shell
  + HBase Shell Command Groups
  + Creating and Populating a Table Using HBase Shell
  + Getting a Cell's Value
  + Counting Rows in an HBase Table
  + HBase Java Client
  + HBase Scanners
  + The Scan Class
  + The KeyValue Class
  + The Result Class
  + Getting Versions of Cell Values Example
  + The Cell Interface
  + HBase Java Client Example
  + Scanning the Table Rows
  + Dropping a Table
  + The Bytes Utility Class
  + Table Schema Main Rules to Follow
  + Good Use Cases for HBase
  + Not Good Use Cases for HBase
  + Business Continuity Caveats
* Introduction to MongoDB
  + MongoDB
  + Main Features
  + MongoDB's Logo
  + Positioning of MongoDB
  + The CAP Placement
  + MongoDB Clients
  + MongoDB Nexus Architecture
  + Blending the Best of Both Worlds
  + What Makes MongoDB Fast?
  + Pluggable Storage Engines
  + The BSON Data Format
  + BSON Caveats
  + MongoDB Terminology
  + MongoDB Data Model
  + MongoDB Data Model (Cont'd)
  + The \_id Primary Key Filed Considerations
  + Indexes
  + (Traditional) Data Modeling in RDBMS
  + Data Modeling in MongoDB
  + An Example of a Data Model in MongoDB
  + MongoDB Data Modeling
  + A Sample JSON Document Matching the Schema
  + To Normalize or Denormalize? Is that a Question?
  + MongoDB Query Language (QL)
  + The
  + find()
  + Method
  + The limit()
  + Method
  + A MongoDB QL Example
  + Query Syntax is Driver-Specific!
  + More Client Code Examples
  + MongoDB Query to SQL Select Comparison
  + Data Inserts
  + Data Lifecycle Management
  + Data Lifecycle Management: TTL
  + Data Lifecycle Management: Capped Collections
  + Data Sharding
  + Data Replication
  + GridFS
  + MongoDB Security
  + Authentication
  + Data and Network Encryption
  + MongoDB Limitations
  + MongoDB Use Cases
* Apache Cassandra
  + What is Apache Cassandra?
  + Main Features
  + Peer-to-Peer (No Master)
  + Wide Column Store NoSQL Databases
  + Cassandra Model vs Relational Model
  + Column Families
  + Columns
  + Simplified Data Model
  + Data Model
  + The Cap Placement
  + CQL
  + CQL Simple Examples
  + The Update Statement
  + Update Caveats
  + Update Statement with TTL and TIMESTAMP Examples
  + Collections
  + Example of Using a Set Collection
  + Using the List Collection
  + Data Replication
  + Visualizing Data Replication
  + The Write Path
  + Sequential Data Storage Engine
  + Java Client Code Example
  + Data Distribution
  + Native Aggregate Functions
  + Creating UDFs
  + HBase vs. Apache Cassandra
  + Cassandra vs. MongoDB
  + Security
  + WAN-Wide High Availability
* Introduction to Hadoop
  + The Client – Server Processing Pattern
  + Apache Hadoop
  + Apache Hadoop Logo
  + Typical Hadoop Applications
  + Hadoop Clusters
  + Hadoop Distributions
  + Hadoop's Main Components
  + Hadoop Distributed File System (HDFS)
  + HDFS Considerations
  + Data Blocks
  + HDFS NameNode Directory Diagram
  + HDFS Balancing
  + Accessing HDFS
  + Examples of HDFS Commands
  + Other Supported File Systems
  + YARN
  + Hadoop-based Systems for Data Analysis
  + MapReduce
  + Similarity with SQL Aggregation Operations
  + MapReduce Word Count Example
  + Distributed Computing Economics
  + Discussion: Divide and Conquer
  + Apache Pig
  + Pig Latin
  + Running Pig
  + Pig Latin Script Example
  + What is Hive?
  + Hive's Value Proposition
  + Who uses Hive?
  + What Hive Does Not Have
  + HiveQL
  + Working with Hive Tables
* Introduction to Functional Programming
  + What is Functional Programming (FP)?
  + Terminology: Higher-Order Functions
  + Terminology: Lambda vs Closure
  + A Short List of Languages that Support FP
  + FP with Java
  + FP With JavaScript
  + Imperative Programming in JavaScript
  + The JavaScript map (FP) Example
  + The JavaScript reduce (FP) Example
  + Using reduce to Flatten an Array of Arrays (FP) Example
  + The JavaScript filter (FP) Example
  + Common High-Order Functions in Python
  + Common High-Order Functions in Scala
  + Elements of FP in R
* Introduction to Apache Spark
  + What is Apache Spark
  + A Short History of Spark
  + Where to Get Spark?
  + The Spark Platform
  + Spark Logo
  + Common Spark Use Cases
  + Languages Supported by Spark
  + Running Spark on a Cluster
  + The Driver Process
  + Spark Applications
  + Spark Shell
  + The spark-submit Tool
  + The spark-submit Tool Configuration
  + The Executor and Worker Processes
  + The Spark Application Architecture
  + Interfaces with Data Storage Systems
  + Limitations of Hadoop's MapReduce
  + Spark vs. MapReduce
  + Spark as an Alternative to Apache Tez
  + The Resilient Distributed Dataset (RDD)
  + Spark Streaming (Micro-batching)
  + Spark SQL
  + Example of Spark SQL
  + Spark Machine Learning Library
  + GraphX
  + Spark vs. R
* The Spark Shell
  + The Spark Shell
  + The Spark Shell UI
  + Spark Shell Options
  + Getting Help
  + The Spark Context (sc) and SQL Context (sqlContext)
  + The Shell Spark Context
  + Loading Files
  + Saving Files
  + Basic Spark ETL Operations
* Spark RDDs
  + The Resilient Distributed Dataset (RDD)
  + Ways to Create an RDD
  + Custom RDDs
  + Supported Data Types
  + RDD Operations
  + RDDs are Immutable
  + Spark Actions
  + RDD Transformations
  + Other RDD Operations
  + Chaining RDD Operations
  + RDD Lineage
  + The Big Picture
  + What May Go Wrong
  + Checkpointing RDDs
  + Local Checkpointing
  + Parallelized Collections
  + More on parallelize() Method
  + The Pair RDD
  + Where do I use Pair RDDs?
  + Example of Creating a Pair RDD with Map
  + Example of Creating a Pair RDD with keyBy
  + Miscellaneous Pair RDD Operations
  + RDD Caching
  + RDD Persistence
  + The Tachyon Storage
* Conclusion