JUPITER PRIME

Consulting • Training • Advisory

جوبيتر برايم للاستشارات والتدريب

COURSE BROCHURE

Advanced Big Data Pipelines and Streaming Analytics Training Course

30 August – 10 September 2027
Amsterdam

Advanced Big Data Pipelines and Streaming Analytics Training Course

Ref: 2537_96395 **Date:** 30 August – 10 September 2027 **Location:** Amsterdam **Fees:** 11100 GBP

Big Data Pipelines and Streaming Analytics is a 2-week training course for data analysts and analytics managers who must turn large, fast and varied data into timely decisions, with each participant producing a Streaming Analytics Pipeline Blueprint. Event volumes from sensors, apps and transactions now outgrow nightly reports, so organisations need analysis that keeps pace with operations. Jupiter Prime Training builds this course around design choices that hold up at scale.

Why Big Data Pipelines and Streaming Analytics Matter Now

Connected devices and payment systems emit events continuously, so operators expect answers in minutes. Cloud storage is cheap, yet compute spend climbs fast when jobs are poorly designed, and auditors expect every streamed figure to be traceable and protected. Well-designed pipelines catch fraud, faults and demand shifts early; weak ones produce stale dashboards and rising bills. The course rests on vendor-neutral distributed computing and stream processing principles, so the designs participants produce carry across whichever cloud or on-premises platform their organisation already runs.

Who Gains Most from Big Data Pipelines and Streaming Analytics

- Analysts building metrics on high-volume event and transaction data
- Analytics leads deciding which questions need real-time answers
- Data quality owners accountable for freshness and lineage of shared tables
- Operations and risk analysts monitoring fraud or equipment signals live
- Analytics managers who balance platform spend against the value of faster insight

An analytics function that sends an analyst, a data quality owner and its manager together gains most, because the three can agree latency targets, monitoring duties and cost limits in one shared design.

Big Data Pipelines and Streaming Analytics Capabilities

Participants leave able to reason about partitions, windows and watermarks as confidently as they reason about averages and filters.

- Assess which decisions need streaming latency and which suit batch processing — analytics leads and budget holders
- Design partitioned, layered storage that keeps large scans fast and affordable — platform and finance stakeholders
- Specify event windows, watermarks and late data rules for real-time metrics — operations managers relying on live figures
- Build quality monitors, data contracts and lineage that expose silent pipeline failures — report consumers and auditors
- Apply sketches, sampling and pre-aggregation to analyse billions of rows quickly — analysts serving executive dashboards
- Weigh access controls, masking and cost attribution before production release — privacy officers and finance partners

Key Big Data Pipelines and Streaming Analytics Terms

- Data pipeline — A sequence of automated steps that ingests, transforms and publishes data for analytical use.
- Event stream — An ordered, continuously growing log of records, each describing something that happened at a point in time.
- Partitioning — Splitting a large table or stream into separate segments by a key so work and reads run in parallel.
- Windowing — Grouping an unbounded stream into finite time slices so counts, sums and averages can be computed.
- Watermark — A moving marker of event time that tells a stream processor when a window can safely be closed.
- Data lineage — A traceable record of where each figure came from and which transformations produced it.

Big Data Pipelines and Streaming Analytics Day by Day

Day 1: Scale, Velocity and Variety in Context

- Volume, Velocity and Variety Profiling — Sizing the Real Analytical Problem
- Batch Versus Streaming Trade-Offs — Choosing Latency That Decisions Actually Need
- Data Product Thinking — Assigning Owners to Shared Analytical Outputs
- Cost-to-Insight Mapping — Linking Compute Spend to Business Questions
- Stakeholder Latency Interviews — Capturing How Fast Each Decision Moves

Day 2: Distributed Storage Foundations

- Object Storage and Data Lakes — Placing Raw Records Cheaply and Durably
- Columnar File Formats — Speeding Scans Across Billions of Rows
- Partitioning and Bucketing Schemes — Pruning Reads for Frequent Queries
- Lakehouse Table Layers — Adding Transactions and Versioning to Lake Files
- Retention and Tiering Rules — Balancing Storage Cost Against Recall Needs

Day 3: Distributed Processing Concepts

- Parallel Execution and Shuffles — Diagnosing Why Large Jobs Stall
- Map, Reduce and Dataframe Operations — Expressing Transformations at Scale
- Skew and Hot Key Handling — Rebalancing Work Across Cluster Nodes
- Memory, Spill and Caching Choices — Keeping Heavy Jobs Within Budget
- Job Profiling and Execution Plans — Finding the Costliest Processing Stage

Day 4: Ingestion and Pipeline Design

- Change Data Capture — Streaming Database Updates Without Full Reloads
- Raw, Cleaned and Curated Zones — Setting Trust Levels
- Idempotent Loads and Backfills — Rerunning Pipelines Without Duplicate Rows
- Orchestration and Dependency Graphs — Scheduling Interlinked Pipeline Tasks
- Schema Evolution Policies — Absorbing New Fields Without Breaking Consumers

Day 5: Event Streams and Messaging

- Publish and Subscribe Event Logs — Decoupling Producers From Many Consumers
- Topics, Partitions and Offsets — Ordering Events and Replaying History
- Delivery Semantics — Choosing At-Least-Once or Exactly-Once
- Event Schema Registries — Versioning Message Contracts Between Teams
- Back-Pressure and Consumer Lag Monitoring — Spotting Falling-Behind Readers

Day 6: Streaming Computation

- Tumbling, Sliding and Session Windows — Aggregating Unbounded Event Flows
- Event Time Versus Processing Time — Correcting for Late Arrivals
- Watermarks and Late Data Policies — Deciding When a Window Closes
- Stateful Stream Joins — Enriching Events With Reference and Profile Data
- Checkpointing and Recovery — Resuming Streams After Node Failure

Day 7: Scalable Analysis Methods

- Approximate Counting and Sketches — Estimating Distinct Users Across Huge Logs
- Sampling Strategies for Large Tables — Keeping Estimates Representative and Fast
- Distributed Feature Engineering — Preparing Model Inputs From Event Histories
- Real-Time Anomaly Scoring — Flagging Fraud, Faults or Demand Spikes
- Pre-Aggregated Cubes and Materialised Views — Serving Dashboards at Speed

Day 8: Data Quality and Observability

- Data Contracts — Defining Expected Shape and Freshness
- Automated Expectation Tests — Blocking Bad Batches Before Publication
- Freshness, Volume and Distribution Monitors — Detecting Silent Pipeline Breaks
- Lineage Graphs — Tracing a Dashboard Figure Back to Its Source Event
- Incident Runbooks for Broken Pipelines — Restoring Trust After Data Failures

Day 9: Security, Privacy and Cost Control

- Column-Level Access Rules — Restricting Who Reads Sensitive Fields
- Masking, Tokenisation and Pseudonymisation — Protecting Personal Fields in Streams
- Data Residency and Retention Constraints — Placing Workloads Within Policy Limits
- Workload Cost Attribution — Charging Compute Back to Consuming Teams
- Autoscaling and Spot Capacity Policies — Trimming Idle Cluster Spend

Day 10: Practising the Streaming Analytics Pipeline Blueprint

- Latency and Storage Choices Applied — Selecting Layers for a Live Case
- Event Window and Watermark Design Applied — Specifying Real-Time Metrics
- Quality Monitors and Lineage Applied — Setting Alert Thresholds and Owners
- Cost and Access Controls Applied — Drafting Guardrails for Production Use
- Streaming Analytics Pipeline Blueprint — Presenting the Design for Sign-Off

Applying Big Data Pipelines and Streaming Analytics After the Course

- Run a latency review of five recurring reports, classifying each as batch or streaming and recording the cost and freshness case for the choice.
- Add freshness, volume and distribution monitors to one critical pipeline, with named owners and a runbook for the first alert.

Big Data Pipelines and Streaming Analytics Questions FAQ

What are big data pipelines and streaming analytics, and why do they matter now?

They are the automated flows that move large, fast data from sources to analysis, and the methods that compute results while events still arrive. They matter now because operational decisions in retail, utilities and finance increasingly depend on minutes-old figures.

Which roles gain most from big data pipelines and streaming analytics training?

Analysts and analytics managers working with high-volume event, sensor or transaction data gain most. Data quality owners and risk analysts who set alert thresholds also benefit.

How does big data pipelines and streaming analytics training differ from a reporting or dashboard course?

It works beneath the dashboard. A reporting course focuses on visual design and business metrics, whereas this course covers distributed storage, parallel processing, event streams, windowing and pipeline observability that determine whether those dashboards stay accurate and fast at scale.

What do you bring back from big data pipelines and streaming analytics training?

You bring back a reviewed design for one live analytical pipeline, covering storage layers, stream windows, quality monitors, access controls and cost guardrails.

Big Data Pipelines and Streaming Analytics: The Way Forward

The Streaming Analytics Pipeline Blueprint first decides which business signal moves from overnight reporting to continuous computation. Analysts, data quality owners and analytics managers use it to agree windows, alert owners and spend limits. After its first month in production, consumer lag, late data rates and cost per query deserve another look, since real traffic reveals where the design needs tuning.