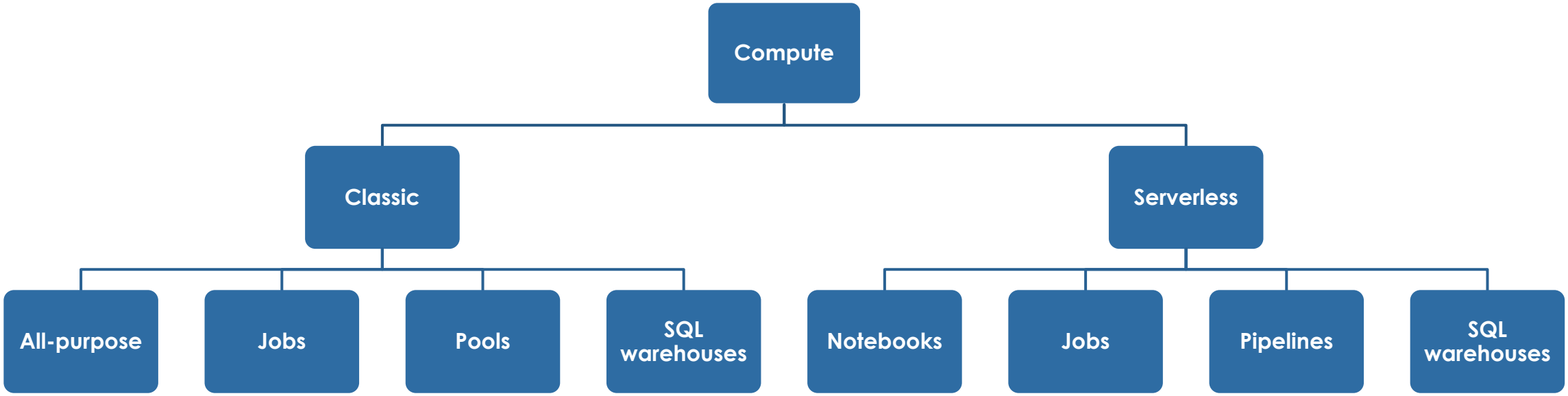
Cluster Best Practices

Learning Objectives

- Compute types
- Best practices and recommendations

Databricks Compute



Classic Compute Types

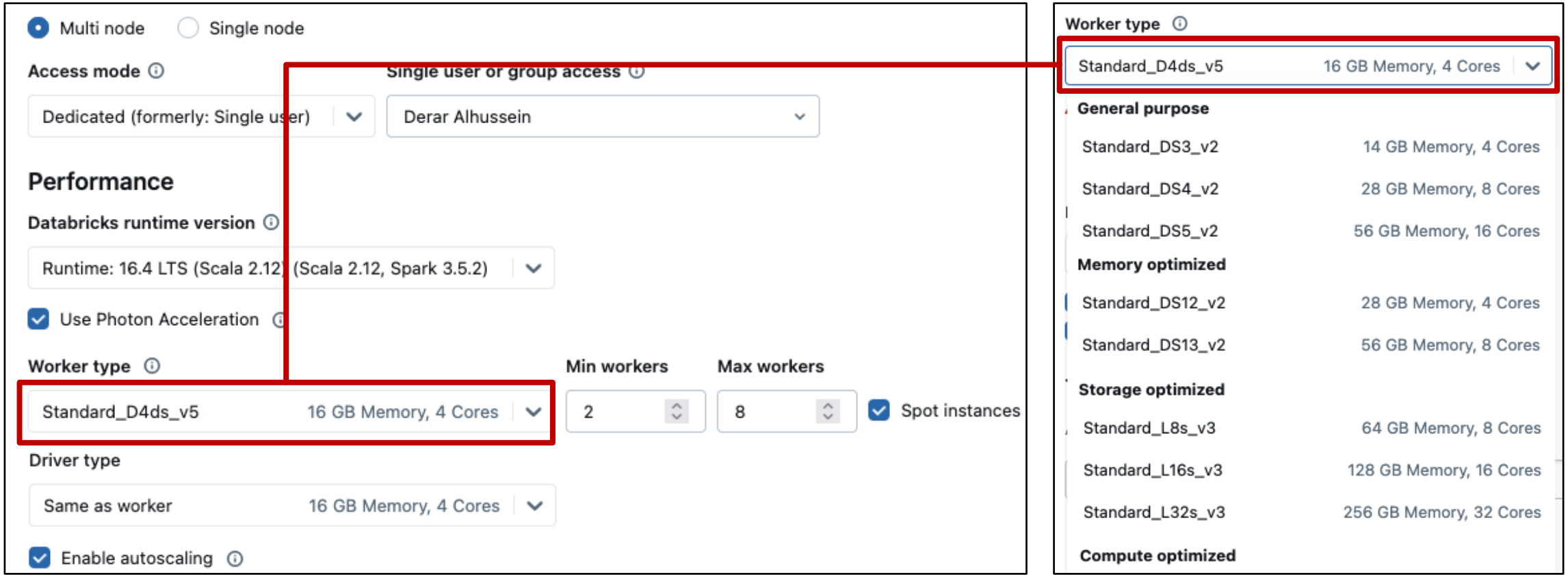
All-purpose cluster

- Used to run interactive notebooks
- Can be created using the Web UI, Command Line (CLI), or REST API
- Can be manually terminated, or auto terminated after a period of inactivity
- DBU: More expensive

Jobs cluster

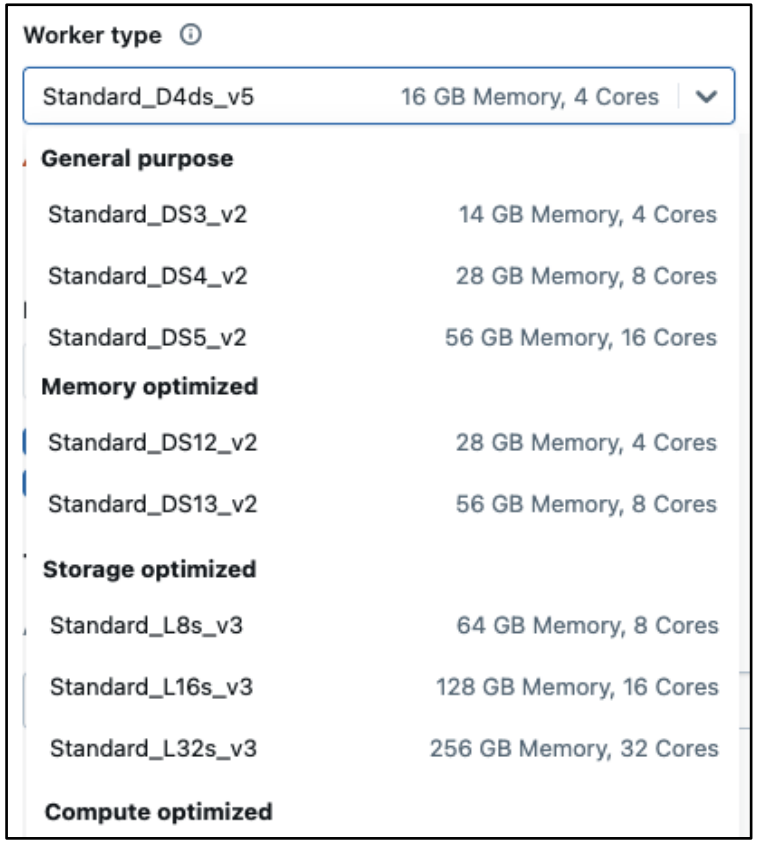
- Used to run automated jobs
- Created automatically by Databricks job scheduler
- Terminate when the job is completed
- DBU: Less expensive

Instance Family



Instance Family

VM Category	Workload Type
Memory Optimized	• Where a lot of shuffle and spills are involved • When Spark caching is needed • For ML workloads
Compute Optimized	• Structured Streaming jobs • ELT with full scan and no data reuse • To run OPTIMIZE and Z-order Delta commands
Storage Optimized	• To leverage Delta caching • For ad hoc and interactive data analysis • ML and DL workloads with data caching
GPU Optimized	• ML and DL workloads with an exceptionally high memory requirement
General Purpose	• Used in absence of specific requirements • To run VACUUM Delta command

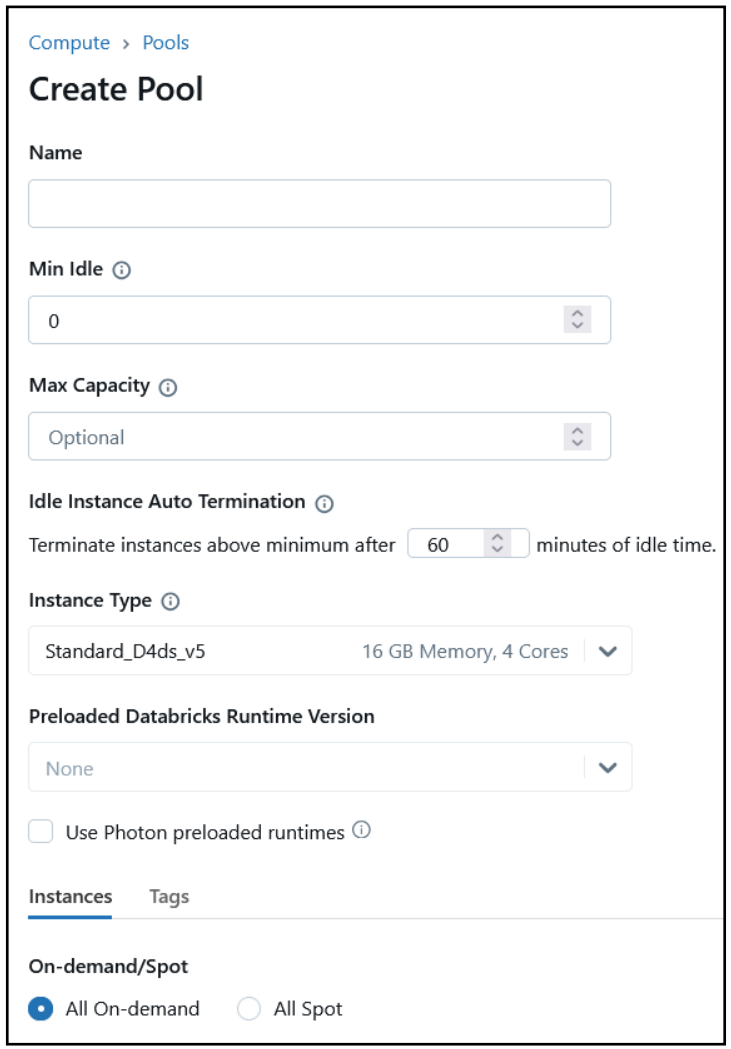


Spot Instances

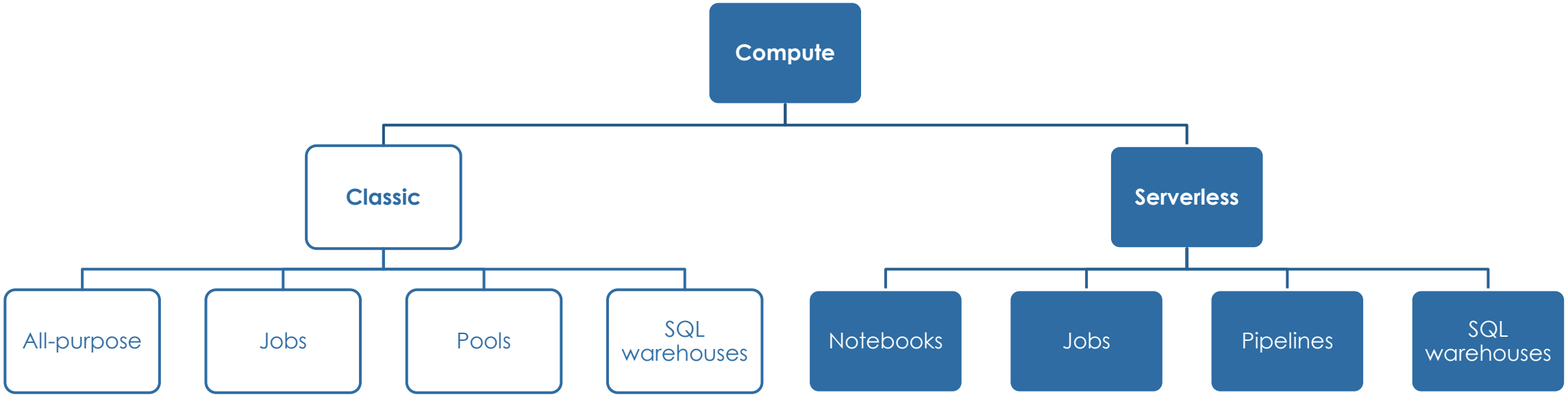
- Low-cost virtual machines
- Users bid for unused capacity, but the instances can be interrupted if the market price exceeds the bid.

Databricks pools

- Set of idle, and ready-to-use instances
- Reduce cluster start and auto-scaling times
- Cloud provider charges

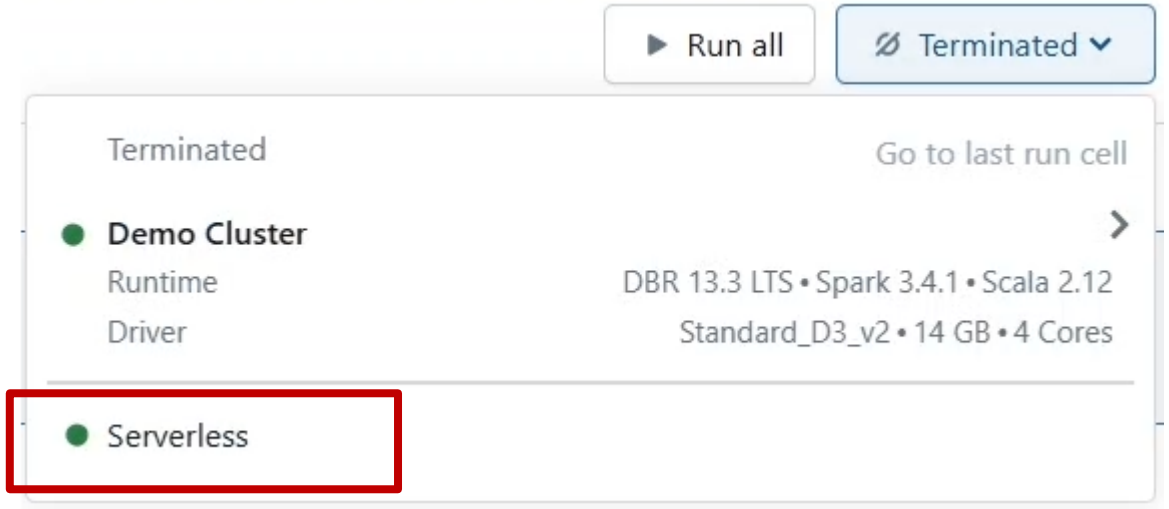


Serverless Compute



Serverless Compute

- On-demand, scalable computing resources without configuration
- Avoid the operational overhead of managing and tuning clusters.
- Rapid start-up and scaling times
- Runs latest Databricks Runtime
- Support only Python and SQL



Serverless Compute Types

- Serverless compute for notebooks
- Serverless compute for jobs
- Serverless compute for pipelines

Serverless vs. Classic

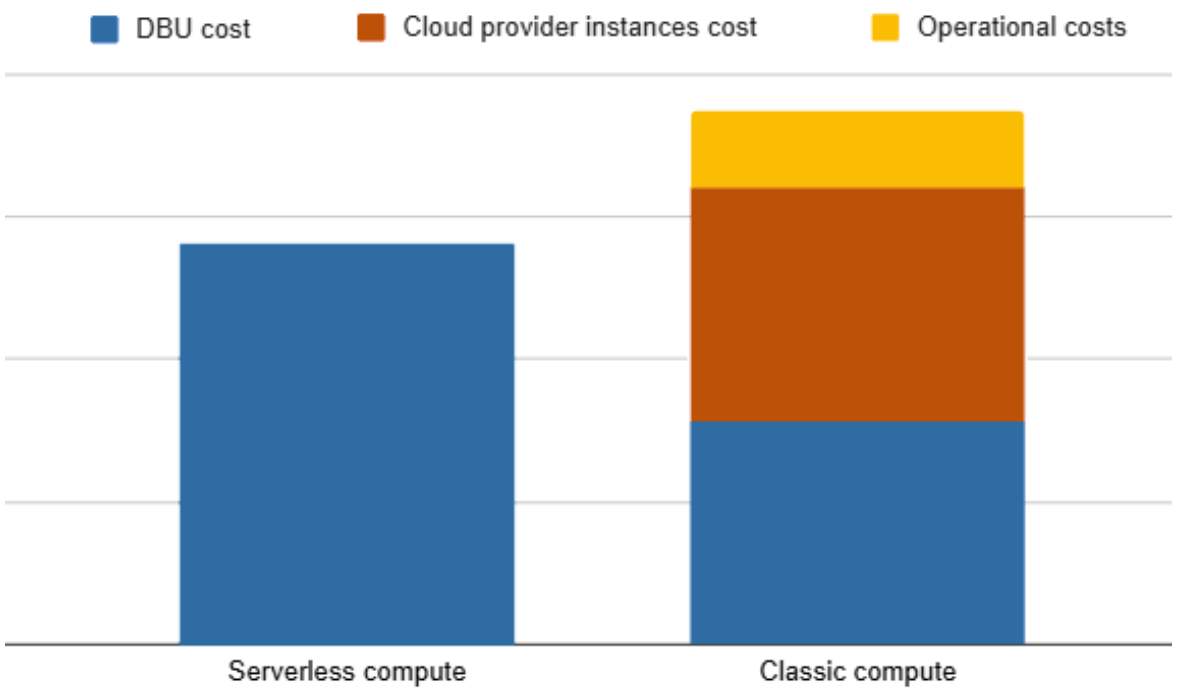
Serverless compute

- Fully managed without configuration
  - Photon Engine is enabled by default
  - Automatic access to latest features
  - Automatic instance type selection
  - Automatic scaling
- Rapid Startup time
- Supports only Python and SQL

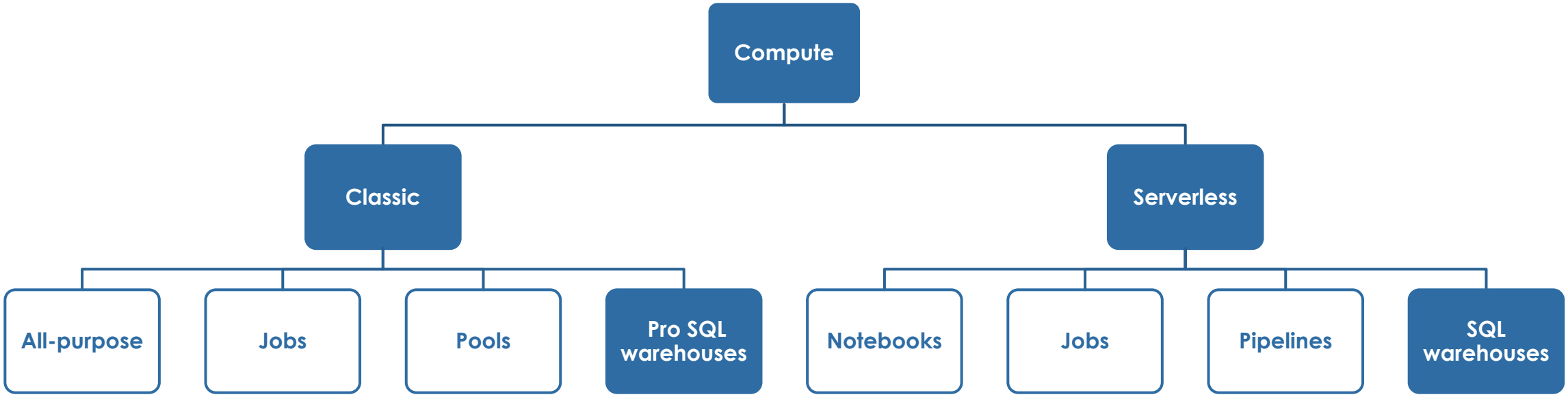
Classic compute

- Offers additional control and configuration
  - Photon Engine is optional
  - Requires manual upgrade of runtime versions
  - Requires manual selection of instance type
  - Autoscaling requires manual configuration
- Takes few minutes to start
- Supports Python, SQL, Scala, R, and Java

Pricing


DBU cost
Cloud provider instances cost
Operational costs
Serverless compute
Classic compute

SQL Warehouse



SQL Warehouse

- Highly optimized for SQL workloads
- Serverless SQL warehouses:
  - ETL
  - Business intelligence
  - Exploratory analysis
- Pro SQL warehouses:
  - If Serverless are not available
  - For custom-defined networking (databases/event buses in your cloud network or on-premises)