VOLT ACTIVE DATA

Active(AI)

Run AI Models Closer to Your Data

Many teams score streaming events by calling a separate inference service. That means a network round trip for every event, another tier to deploy and scale, and a redeployment whenever the model changes.

When the model is far from the data, the consequences are immediate:

- A Network Call per Event
- A Separate Inference Tier
- Redeploy to Update a Model
- Format Lock-In
- Two Systems to Operate

Active(AI) is the set of VoltSP plugins that brings model inference into the pipeline itself. The ONNX processor runs a model on each event as it passes. The ONNX GenAI processor runs generative models. The MLflow model resource watches an MLflow registry and hands new model versions to the running pipeline.

How it works

The ONNX processor loads a standard ONNX model and runs it on each event. A tensor extractor, written in Java or as a JavaScript function in YAML, turns the event into the model's input tensor. A compute function then builds the outgoing event from the original input and the model's outputs.

Models load from local disk or, through VoltSP's storage plugins, from an S3-compatible bucket, and are cached locally with configurable size and expiry. The ONNX GenAI processor does the same for generative models.

The MLflow model resource connects a pipeline to an MLflow tracking server. It watches one registered model, either a named stage such as Production or the latest version, polling every 60 seconds by default. When a new version appears it downloads it and notifies the ONNX processor, which reloads the model without restarting the pipeline.

Inference is one step in an ordinary VoltSP pipeline. Events arrive from any VoltSP source, can be prepared by Java, JavaScript or Python processors, and the scored result can go to any VoltSP sink, including a Volt stored procedure.

What this means in practice

Score each event where it is processed. The model runs inside the VoltSP worker, so scoring adds no network call to a separate service. Inference capacity grows as the pipeline scales out.

Promote a model in MLflow and the pipeline picks it up. With a stage configured, moving a version to Production is all it takes; the pipeline loads it within one polling interval. Rolling back is moving the earlier version back into that stage.

Bring models from the tools you already use. Any framework that exports to ONNX can supply the model, and MLflow remains the registry of record. Data science keeps its tooling; the platform team keeps its pipelines.

From inference service to inference in the stream

	Separate Inference Service	Active(AI) in VoltSP
Where scoring happens	A separate service, one network call per event	Inside the pipeline worker, in the same process
Model updates	Redeploy the service or the pipeline	MLflow resource detects a new version; the processor reloads it without a restart
Model format	Depends on the serving stack	ONNX, from any framework that exports it; ONNX GenAI for generative models
Hardware	A dedicated inference fleet	CPU by default; CUDA execution provider optional
Scaling	Scaled separately from the pipeline	Scales with the pipeline

Architectural properties

- **ONNX processor:** standard ONNX models
- **ONNX GenAI processor:** chat templates, generation parameters, optional token-by-token streaming
- **Model sources:** local disk, or an S3-compatible bucket through VoltSP's S3 plugin; local cache with size limit, expiry and clean-up on start
- **ONNX Runtime session options:** intra- and inter-op threads, graph optimization level, execution mode, CUDA execution provider and device, CPU execution provider (default)
- **MLflow model resource:** tracking URI, model name, optional stage (otherwise latest version), poll interval (default 60 seconds), auth token, cache directory; notifies the processor, which hot-reloads the model
- Configured in VoltSP's Java DSL or YAML, alongside VoltSP's sources, sinks and Java, JavaScript and Python processors

Key outcomes

- Score events in the pipeline, without a network call to a separate inference service
- Put a new model version into production by promoting it in MLflow, with no pipeline redeploy
- Keep data science on its own tools: any framework that exports ONNX, with MLflow as the registry
- Choose CPU or GPU per pipeline through configuration

AVAILABILITY

Active(AI) is available now in VoltSP 1.8 and is licensed as the Active(AI) feature. It requires VoltSP, licensed as Active(SP), and makes full use of a Volt cluster for stateful features and decisions.

Book a technical call voltactivedata.com/contact

Volt Active Data is the real-time decisioning layer for mission-critical systems.