What does SAS® Event Stream Processing do?
SAS Event Stream Processing analyzes and understands millions of events per second, detecting patterns of interest as they occur. The results show the correct actions to take, what alerts to issue, which data to store and which events to ignore.

Why is SAS® Event Stream Processing important?
Get value out of streaming data from operational transactions, sensors, devices, transmissions, etc. – before it’s stored. With submillisecond response times, the software assesses data streams using a suite of prebuilt operators, functions, routines and advanced analytics. So you can derive insight into events and take appropriate actions.

For whom is SAS® Event Stream Processing designed?
A configurable, embeddable engine, the software is designed for application developers, IT engineers and system architects. The web-based interface makes it easy to manipulate data and build projects, while a dashboard delivers information visually so you can test and validate results.

SAS® Event Stream Processing
Act on big data while it’s in motion to keep a real-time pulse on your business

Staying competitive in a big data world means working and making decisions incredibly fast. You need to assess continuously changing conditions, monitor for events of interest as they happen and follow through with contextually relevant action. With event stream processing, the data doesn’t stop. Continuous queries constantly analyze live data to detect patterns of interest so you can respond in real time.

SAS Event Stream Processing helps you understand events while they are in motion (known as event streams). Instead of the traditional “stream, score and store” model, queries are stored and data is streamed through them. Data is analyzed continually as it’s received, updating the situational intelligence as new events take place. Value, which otherwise would be lost through information lag, is captured immediately. The solution is designed for optimal performance with a flexible processing model that provides submillisecond response for high-volume throughput.

Benefits

- **Get instant information, take immediate action.** Streaming data from operations, transactions, existing systems, and Internet of Things (IoT) sensors and devices is valuable when you can act upon key insights in real time. Regardless of data or format, SAS immediately enhances the value of streaming data for information-driven applications, including customer experience, asset and IT performance, fraud, risk, security and more.

- **Store the right big data.** Faster, better and more powerful stream processing ensures data corrections are made before high-volume throughput data is stored – reducing big data management processing costs. Continuous queries run faster in the flexible threaded processing model, which natively supports dynamic updates, deletions and insertions while data is in motion.

- **Make sound decisions based on sound analyses.** Having accurate data at your disposal means you can take the right action. SAS Event Stream Processing provides in-stream, prebuilt data quality operators, natural language text-processing functions and a wealth of advanced analytic algorithms to detect even the most complex event stream patterns.

- **Take full control and adapt to changes quickly.** No matter how fast your data moves, how much data you have, or how many data sources you’re connecting to, it’s all streamlined and under your control from a single, intuitive interface. Consolidated event stream data instructions defined in SAS and other languages ensure streamlined, governed pattern detection, which makes it easier to evolve with changing data conditions.
Overview

SAS Event Stream Processing ingests large volumes of streaming data quickly – millions of events per second – so you can understand events in the data as it’s being generated and before it’s stored. No stream of data is too big or fast. The solution is designed for exceptional processing speeds, bound only by the hardware environment’s limitations.

Incoming data is read through adapters and connectors, which are part of a publish-and-subscribe architecture. Event data publishes into a source window of an event stream processor. A visual interface makes it easy to define the windows, procedures and operators. In turn, it’s simple to define continuous queries through which the data will stream. Streaming data is examined for patterns and can be intelligently filtered to store anomalies that demand deeper investigation. Or, if no relevancy is detected, the data can be discarded. Downstream applications subscribe to receive streaming analysis results with prescribed actions.

Consume and connect streaming data

Is your data center prepared to handle the rapid growth projected for the Internet of Things data? Most are not. Even data streams generated from current web transactions, sensor devices and operational systems can be a challenge to consolidate and fully exploit. SAS Event Stream Processing has the power to consume a wide array of data stream formats, and can easily integrate results into existing systems and applications. An extensive suite of prebuilt adapters and connectors enables you to source and manipulate live data streams to detect, filter, aggregate, correlate and delineate patterns of interest. Actions are applied to both structured and unstructured text data, with results sent to other systems to improve efficiency and lower costs. Your organization can consume the data sources it needs today with assurance that the same technology will work tomorrow.

Flexible, visual development environment

SAS Event Stream Processing provides a highly visual, interactive interface for building modular, continuous queries that use SAS advanced analytics algorithms and rules to pinpoint event relevance. System architects and application developers have point-and-click access to the palette of windows and connectors, making it easy to design complex streaming analytical models. An interactive test mode lets users evaluate logic and validate results before deployment. Events can be shared across projects and historic activity can be retained and compared to current events – all from the visual drag-and-drop environment. This means data in motion can be easily defined, updated and revised to address new and quickly emerging events.

Advanced pattern-matching algorithms

Data processing within streams has never been more thorough. SAS Event Stream Processing provides parsing, filters, joins, field calculations and pattern-matching functions available right out of the box. This includes prebuilt functions specifically designed for routine data management tasks, like transformations, normalization, matching, identification and more. You can also edit prebuilt functions and definitions of custom functions to further increase flexibility. Unstructured text processing functions provide natural language processing (NLP) extractions of concepts, entities and facts, as...
well as classification of text and sentiment identification. In-stream clustering builds homogeneous groups of events on the fly while predictive algorithms estimate future events, creating advanced patterns of interest. You can examine multiple events in a single query, including both sequential and time-based (temporal) events.

**Visual investigation of event streams**
Understanding conditions and situational status is critical. Streaming visualization of alert triggers and notifications provides an immediate understanding of streaming data actions. You can validate event stream activity before deploying into production with a browser-based stream viewer. This configurable dashboard allows you to easily and quickly discern what is within acceptable thresholds and what is trending outside of desired norms. The technology also depicts alerts and notifications as part of pattern detection. When you need to understand live conditions, streaming visualization testing gives you an immediate and easily recognizable picture of real-time behavior.

**Optimized processing that scales**
With event stream processing, systems need to be online, all the time. And processing must happen very fast, which demands optimal performance. SAS has patented, instantaneous 1+N way failover, guaranteed delivery without persistence and dynamic updates to ensure consistent, relevant processing. The low latency, high-volume SAS Event Stream Processing software outperforms other stream processing engines with commodity hardware. And the distributed, in-memory grid processing scales linearly as your data grows, optimizing hardware investments.

**Flexible, threaded processing model**
For additional performance gains, take advantage of highly modular, continuous queries that support customizable, project-based threading to process alternate scenarios differently. Production performance can be set differently from testing performance, and you can slow performance to gain in-depth evaluation when testing scenarios. A mixed-processing mode

---

**Key Features**

**Consume and connect streaming data**
- Extensive suite of data adapters and connectors for publishing and subscribing to live data streams of both structured and unstructured data.
- Predefined adapters include read and write (i.e., publish and subscribe): Hadoop (HDFS Yarn), Apache Camel, OSIsoft PI, Axeda, Rabbit MQ, Solace, Tervela Data Fabric, XML/JSON File Socket Adapter, SAS’ LASR™ Analytic Server, IBM DB2, IBM Netezza, IBM WebSphere MQ, SAP Sybase ASE, Tibco Rendezvous, JMS, File/Socket, Database ODBC and SAS data sets.
- Customized publish/subscribe APIs can also be written in C or Java.
- Publish only from the following: Twitter, Log Sniffers (Oracle, Greenplum), Network Sniffer, Google Protobuf Sniffer, SYSLOG, HTTP RESTful interfaces; subscribe only to SOAP and SMTP.
- Event handlers are used to manage multiple input streams.
- Persist and fully synchronize event streams using features in guarantee mode or via the publisher/subscriber API to trigger persistence of an event stream.

**Adaptable, in-stream analytics and data manipulation**
- Rich, graphical interface for point-and-click design and testing of event logic. Comparable C++ library and XML syntax for event model definition also available:
  - Full visibility to the automatically generated XML code, which can be customized.
  - Visually defined event stream processing models with stateful indicators are easily instantiated to the XML server, connecting live data streams for model validation.
- Flexible, modular, window-driven architecture to define complex continuous queries:
  - Based on an extensive suite of interchangeable window types and operators, to detect an unlimited number of patterns, correlations, computations and aggregations.
  - Operators include procedural, declarative, and rules and custom procedures defined with C++, DS2 or SAS DATA step callouts.
  - Continuous queries are defined as a series of window types chained together (graphically or programmatically).
  - Prebuilt, common data quality routines are used to cleanse, standardize and filter live stream data before it’s stored, reducing downstream processing.
  - Patterns of interest can include virtually unlimited advanced analytics calculations, with both in-stream machine learning K-means clusters and live stream analytical scoring. Analytical models include all SAS DATA step, DS2 and third-party code snippets.
  - Event stream assessments are adapted with dynamic updates to active stream processing for inserts, deletes and replacement.
  - Highly embeddable to gateways, devices and any existing C++ application (with dedicated thread pool processing).

**Visual monitoring of event streams**
- Configure the dashboard for customized testing of streaming activity.
- Isolate data cleansing routines and monitor specific attributes of interest.
- Interactively filter and query live stream activity to examine specific behavior of elements.
- Define and customize notifications by SMS, email and other alerts as part of event stream model workflow.
- Compare historical activity with current processing using graphical representations.
- Monitor stream processing detail by subscribing to events of interest.
can be used for multiple projects to reach desired performance specifically tailored to business scenario needs. And thread pool size can be tuned to refine project processing speeds. You can also easily configure different levels of determinism to test and assess event stream processing without modifying the processor structure.

Enterprise administration and management

Centralized administration and management of event stream processing activities are key to enforcing IT command and control of project handling, processing optimization and maintenance. The solution provides transparent governance that aligns to business requirements and can integrate stream processors and algorithms from SAS and other sources. An optional caching store is also available if data sizes exceed the available memory.

Extend other SAS® solutions with streaming analytics

SAS Event Stream Processing uses results of other SAS solutions such as SAS Customer Intelligence, SAS Asset Performance Analytics and SAS Security Intelligence. These domain-specific applications can be used to identify patterns of interest to embed in SAS Event Stream Processing. Anomalies can be flagged in stream and sent back to SAS solutions for in-depth identification, defining new patterns of interest to monitor. Multistage analytics ensures constant monitoring with continuous improvement for an integrated analytical cycle.

To learn more about SAS Event Stream Processing system requirements and see other related material, please visit: sas.com/ESP.

Key Features (continued)

In-memory, distributed and optimized processing that scales

- High-volume processing of millions of events per second. Low latency response times (millisecond, submillisecond).
- All retained and aggregated data is kept in memory in indexed stores, with indices to each event field. Joins are permitted between primary and secondary indices.
- In-memory processing is optimized with pattern compression of partially completed assessments.
- The pipeline processing architecture and flexible threaded processing model take full advantage of distributed grid architectures.
- Processing speeds can be customized with flexible thread pool sizing, caching stores and more.
- Without modifying the processor structure, easily configure different levels of determinism and set different rates for development, test and production execution.
- Includes patented, instantaneous 1+N way failover, native failover, guaranteed delivery without the use of persistence and other fault-tolerance functions to ensure successful event stream processing activity.
- Full and open access to all event metadata.

Enterprise administration and management

- RESTful interface to XML factory server makes it easy to introduce multiple projects, validate XML syntax, control analysis windows and define project retention policies.
- Centralized multiproject management eases administrative burdens, setting controls to dynamically load projects, start/stop/remove and persist projects.
- Can include authentication for TCP/IP client for both pub/sub and HTTP connections.
- Configuration files ease initial setup protocol specifications.
- Customizable retention strategy based on size, volume or time options associated with policy definitions.
- Enhanced error handling to increase the resilience of handling run-time errors.
- Localized log messages.
- Ability to investigate processing activity with queries into live event streams of interest.
- Authentication and encryption for events streaming in and out to other systems.
- XML syntactic checks use prebuilt validation scripts to automatically generate error messages.
- Manage processing performance with configured threaded processing definitions specific to each project.