SAS® Event Stream Processing 6.2: What's New

What's New for SAS Event Stream Processing 6.2

Release 6.2.7 (March 2024)

Changes to Operating System Support and Software Support

SAS Event Stream Processing 6.2 now supports Red Hat Enterprise Linux 8.4 and later on OpenSSL 1.1. Access to RHEL7 and RHEL8 distributions are available, giving you a choice for which one to use.

Updated Database Drivers

The versions of ODBC drivers for the databases that are supported by the Database connector and adapter have been updated. For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.6 (December 2023)

Version of H2 Database Used by Metering Server Has Been Updated

The version of the H2 database that is used by the SAS Event Stream Processing metering server has been updated from 2.1 to 2.2. Delete your existing database in order to re-create it at the appropriate version.

Updated Database Drivers

The versions of ODBC drivers for the databases that are supported by the Database connector and adapter have been updated. For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Additional Data Types Supported by the OPC-UA Publisher Connector and Adapter

The OPC-UA publisher connector and adapter now support the following data types:

- BOOLEAN
- FLOAT
- array of FLOAT
- GUID
- array of GUID
- INTEGERID
- STRING
- UINT32
- array of UINT32
- UINT64
- UTCTIME

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

New Header Parameter for Kafka Subscriber Connector and Adapter

The Kafka subscriber connector and adapter now enable you to specify a comma-separated list of window schema field names. The CSV representation of the contents of those fields is copied to Kafka message headers. For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Changes in Publish/Subscribe API Support for Apache Avro Messaging

Publisher support for AVRO_BYTES (logical decimal type only) to ESP_DOUBLE type has been added. Also, the AVRO_ARRAY type is processed differently from previous releases. For more information, see *SAS Event Stream Processing: Publish/Subscribe API Reference*.

New Configuration Parameter for Guaranteed Delivery in the Publish/Subscribe API

The Publish/Subscribe API now provides the `GDpub_numAcks` configuration parameter to set the number of acknowledgments that are received from subscribers in order to generate a positive acknowledgment. For more information, see *SAS Event Stream Processing: Publish/Subscribe API Reference*.

Release 6.2.5 (March 2023)

Changes to Publish/Subscribe API Support for Apache Avro

Publisher support has changed in the following ways:

- AVRO_BYTES, AVRO_ENUM, and AVRO_MAP types are now supported.
- Conditional support for AVRO_ARRAY and AVRO_UNION has been updated.
- The JSON representation of an AVRO_MAP can be written to a string field in SAS Event Stream Processing.
- You can no longer write the JSON representation of an AVRO_UNION to a string field in SAS Event Stream Processing.

For more information, see "Publish/Subscribe API Support for Apache Avro Messaging" in *SAS Event Stream Processing: Publish/Subscribe API Reference*

Enhancements to Connectors and Adapters

- The PI Web connector and adapter now provide enhanced error logging to show failed PI points in response to OMF container requests. For more information, see "Using the PI Web Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.
- The Kafka publisher connector and adapter now provide the `avronestingseparator` parameter. For more information, see "Using the Kafka Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.
- The RabbitMQ publisher connector and adapter now provide the `prefetchcount` parameter. For more information, see "Using the RabbitMQ Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.4 (October 2022)

Support for Blob Format Added to C and Java Libraries

C and Java protobuf libraries now support data in blob format. For more information, see *SAS Event Stream Processing: Publish/Subscribe API Reference*.

Miscellaneous Enhancements to Connectors and Adapters

- The OPC-UA connector and adapter have been enhanced as follows:
  - The UA_XVType and the OPC-UA abstract types are now supported.
  - The `opcuasecuritymode` and `opcuasecuritypolicy` parameters were added.
- The PI Web connector and adapter have been enhanced as follows:
  - The `nocachecontrolheader` parameter was added to the subscriber.
  - OSIsoft Message Format (OMF) is now supported.
  - The subscriber now bundles requests for element or attribute Web IDs into one batch per event block.
  - The connector and adapter now continue when batch requests for element or attribute Web IDs fail.
- The RabbitMQ connector and adapter have been enhanced as follows:
  - Quorum queues are now supported.
  - The `deadletterexchange` and `deadletterroutingkey` parameters were added.
  - When the RabbitMQ publisher adapter receives a guaranteed delivery NACK from the subscriber adapter, it now sends a corresponding NACK (with `requeue=false`) to the RabbitMQ server.
- The REST adapter has been enhanced as follows:

  - The `maxqueuesize` parameter was added.
  - The `intelligentdecisioningformat` parameter was added to enable requests in SID format.
- Java adapters now set the Java truststore equal to the value of the environment variable `SAS_TRUSTED_CA_CERTIFICATES_JKS_FILE` when defined.

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.3 (October 2021)

Changes to Operating System Support

Release 6.2.3 is the last to support custom Linux systems developed with the Yocto Project.

File and Socket Connector Now Supports ASTERIX Category 240

You now can use the file and socket connector and adapter to stream radar video transmissions that comply with EUROCONTROL Specification for Surveillance Data Exchange ASTERIX Category 240. For more information, see "Using the File and Socket Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*

Miscellaneous Enhancements to Connectors and Adapters

- The CAS adapter can now load tables from a SASHDAT file when it is present.
- For the Database connector and adapter, Oracle log miner mode (enabled through the `oraclelogminer` parameter) now publishes the row ID with each row.
- The `doubleprecision` parameter of the file and socket subscriber connector can now control the number of precision digits printed for ESP doubles in JSON output.
- The Kafka connector and adapter have been enhanced as follows:
  - The `fieldtokafkakey` parameter of the subscriber now enables **`avro`**, **`opaquestring`**, and **`csv`** formats.
  - Saving Kafka message offset and partition to Source window fields is now supported.
- The MQTT subscriber connector and adapter now writes MQTT messages to a dynamic topic, matching the contents of the event field configured in the `mqtttopic` parameter.
- Encryption support was added to the OPC-UA connector and adapter in order to improve user name and password authentication.
- The PI Web connector and adapter have been enhanced as follows:
  - A new `searchfullheirarchy` parameter was added.
  - POST requests by the subscriber are now enabled to implement bulk writes. This is intended to improve performance.

- The QuasarDB connector and adapter have been enhanced as follows;
  - A new `deduplicate` parameter was added. The `inrangestartrel` parameter was applied to every select operation.
  - An IN RANGE clause is now inserted before WHERE or other further clauses.
- Three new optional parameters, `respopaquejson`, `respuseinsert`, and `bufsize` were added to the Rest adapter.
- The `unit` parameter of the Timer connector and adapter now supports a value of **`millisecond`**.

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.2 (December 2020)

Support for the Kafka Sink Connector

SAS Event Stream Processing now provides a sink connector that you can run in a Kafka Connect environment. You can use this sink connector to publish a stream that is received from a Kafka Connect source connector into an event stream processing engine. For more information, see *SAS Event Stream Processing: Using SAS Event Stream Processing with Other Applications*.

New QuasarDB Connector and Adapter

A new QuasarDB connector and adapter publishing to and subscribing from QuasarDB, a distributed database for time series data that combines in-memory capabilities with long-term storage. For more information, see "Using the QuasarDB Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters* .

Miscellaneous Enhancements to Connectors and Adapters

- The Kinesis publisher connector and adapter now support reshard handling.
- When logging a fatal error, the MQTT publisher connector and adapter and the PI subscriber connector and adapter now stop rather than continue to run.
- The PI Web connector and adapter have been enhanced as follows:
  - A new `piauthmethod` was added.
  - Nesting in the `afrootelement` parameter is now supported.
- A new `buspersistencenonexclusive` parameter was added to the RabbitMQ connector and adapter.
- A new `tokeneventfield` was added to the Rest adapter.

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.1 (June 2020)

New Function for the Functional Window

The `isFailoverStandby` function was added to the Functional window. For more information, see *SAS Event Stream Processing: Using Source and Derived Windows*.

Enhancements to the Pattern Window

Optimization of Pattern Window Performance

The Pattern window was enhanced to improve performance. Debugging capabilities at the trace level were added to the `DF.ESP.WINDOW.PATTERN` logger.

New Attribute to the Pattern Element

The `pattern` element now has the `cross-index-timeout` attribute, which affects the temporal condition of open patterns. For more information, see *SAS Event Stream Processing: XML Language Reference for Event Stream Processing Models*.

New Audio Connector and Adapter

A new audio connector and adapter is now available in order to grab digitized audio from a locally installed sound card for publish operations to a Source window. For more information, see "Using the Audio Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*

New PI Web Connector and Adapter

A new PI Web connector and adapter supports publish and subscribe operations for a PI AF (Asset Framework) server. For more information, see "Using the PI Web Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.

Miscellaneous Enhancements to Connectors and Adapters

- A new `republishmodifiedfile` parameter was added to the File and Socket connector and adapter publisher.
- The Kafka connector and adapter have been enhanced as follows:
  - A new `zookeeperloglevel` parameter was added.
  - The `kafkaglobalconfig` and `kafkatopicconfig` parameters now support embedded semicolons.

- The Kinesis connector and adapter now supports blob format (through opaque binary files).
- A new `includecurrentvalue` parameter was added to the PI connector and adapter publisher. A new optional `cachetargetattributes` parameter was added to the PI connector and adapter subscriber.

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Release 6.2.0 (June 2019)

Cloud-Native Capabilities

SAS Event Stream Processing 6.2 provides pre-built Docker images of the following applications:

- the ESP operator
- the ESP server
- the metering server
- SAS Event Stream Processing Studio
- SAS Event Stream Processing Streamviewer
- SAS Event Stream Manager

After you download these images through your Software Order Email (SOE), you can use tools in a GitHub project to develop, deploy, and test an open ESP server in a Kubernetes cluster. Those tools consist of a set of scripts, YAML template files, and sample projects (XML files) that you can run in the ESP server.

For more information about deploying the product in a Kubernetes cluster, see the *SAS Event Stream Processing on Linux: Deployment Guide*.

Enhancements to Streaming Analytics

New Online Models

The following online models are now available:

- Kalman Filter
- Slice Operations
- Smoothing

Added Support for Models That Are Stored in Analytic Store Files

Analytic store files that apply the following algorithms can now be used with SAS Event Stream Processing:

- Association Rule Mining

- Hidden Markov Model
- Kernel Principal Component Analysis (KPCA)

For more information, see "Loading Models Stored in Analytic Store Files " in *SAS Event Stream Processing: Using Streaming Analytics*.

Improvements to the Image Processing Algorithm

- Three types of coordinate specifications are now supported: COCO, RECT, and YOLO. Previously, only RECT was supported.
- Cropping parameters can now be received as part of the event stream. Previously these parameters had to be specified in the model file, and thus all images had to be cropped identically. They now can be supplied from a Score window that uses a deep learning algorithm in an analytic store file. The input map of the Calculate window can now accept the streaming cropping parameters.

For more information, see "Processing Image Data" in *SAS Event Stream Processing: Using Streaming Analytics*.

Added Support for Deep Learning Functionality

The following deep learning functionality that is available in SAS Visual Data Mining and Machine Learning is now supported for use with SAS Event Stream Processing:

- Mask R-CNN for example, segmentation
- a new ROI (region of interest) alignment layer that is used in Mask R-CNN
- Layer Normalization
- a new embedding loss layer to train Siamese, triplet, and quartet networks
- a Survival layer type for the addLayer action (in the deepLearning action set) in order to create deep survival analysis analytic models
- the Cosine Margin Softmax loss function
- custom FCMP functions that you can use to define early stopping criteria during training
- data augmentations (such as cropping, mutating, and flipping) for object detection and segmentation tasks
- deep clustering unsupervised learning applications

Miscellaneous Improvements

- The Streaming Summary algorithm has a new alpha parameter and new output parameters.
- The Lag Monitoring algorithm has a new parameter to specify the correlation type. You now can specify a distance correlation as well as a Pearson correlation.
- The Recommender Model algorithm now supports Non-Negative Matrix Factorization (NMF) in the Train, Model Reader, and Score windows.

Enhancements to SAS Event Stream Processing Studio

Enhancements and changes to the user interface include the following:

- A new layout theme has been implemented to improve the user experience.
- The user interface for test mode is enhanced. For example, in the **Logs** pane, you now can view more than one type of log message simultaneously. In addition, you now can more easily identify your model's window and field types in test mode.
- You now can validate expressions that you enter from the user interface. This functionality validates the syntax of the code that you enter. It can be particularly useful when you enter several lines of codes or paste code from an external source
- Windows that you have selected are displayed in the workspace with a dashed bounding box, making them easier to identify.
- You can now select multiple projects, engines, and ESP servers simultaneously, which enables you to delete them more efficiently.
- The **Engine Definitions** page is redesigned to improve usability.

SAS Event Stream Processing Studio has been enhanced with cloud-native capabilities, which enables you to use a containerized version of the application.

- You now can use the application in a Kubernetes cluster, which makes it easier to deploy and run ESP servers in the application.
- When a project is created in a Kubernetes cluster, an ESP server is created on demand. When the project is stopped, the ESP server is deleted from the cluster.

When using a Pattern window to detect events of interest (EOIs), you now can import your own Pattern logic templates and use them in SAS Event Stream Processing Studio.

For more information, see *SAS Event Stream Processing: Using SAS Event Stream Processing Studio*.

Enhancements to SAS Event Stream Processing Streamviewer

- The application layout theme is changed to improve the look of the user interface.
- The user interface is enhanced with new and clearer icons to improve usability.
- SAS Event Stream Processing Streamviewer is enhanced with cloud-native capabilities enabling you to use a containerized version of the application.
  - You now can use the application in a Kubernetes cluster, which makes it easier to deploy and run ESP servers in the application.
  - You now can visualize the event streams of a project where the ESP server runs within a Kubernetes cluster.

For more information, see *SAS Event Stream Processing: Using SAS Event Stream Processing Streamviewer*.

Pylon Connector and Adapter Now Supports a USB Attached Camera

The Pylon connector and adapter now enables you to communicate with a USB attached camera in order to continuously publish captured frames into a Source window. For more information, see "Using the Pylon Publisher Connector and Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.

New OPC-DA Publisher Adapter

A new OPC-DA publisher adapter can be used for communication with industrial equipment and systems for data collection and control. OPC Data Access (OPC DA) is precursor to OPC Unified Architecture (OPC UA). For more information, see "Using the OPC-DA Publisher Adapter" in *SAS Event Stream Processing: Connectors and Adapters*.

Miscellaneous Enhancements to Connectors and Adapters

- The Avro format library now supports the conversion of an Avro decimal logical type to an ESP_MONEY datavar. The Avro decimal logical type represents an arbitrary precision signed decimal number. The ESP_MONEY datavar is a 192–bit fixed decimal.
- The Cassandra adapter now supports ESP blob fields.
- The database connector and adapter now supports the SQL_BINARY format.
- Added an ESP publish/subscribe server configuration parameter to bind to a specific NIC.
- The file and socket subscriber connector and adapter now supports the following:
  - opaque string data
  - opaque binary files
- The JMS adapter now supports copying MapMessage to and from an ESP blob field.
- Added support for configurable consumer groups to Kafka publisher connector and adapter.
- Added an argument to suppress the event opcode from outbound Avro messages in the Kafka subscriber, Kinesis subscriber, and Kafka client transport.
- Enhanced the Kafka publisher connector and adapter and client transport to honor the configured offset when a consumer group is configured.
- The Kinesis connector and adapter now supports Assume Role credentials.
- The PI publisher connector and adapter now support copying all values to an ESP string field.
- A new parameter to the Pylon connector and adapter is available to compress images in YUV422_YUYV_Packed format into JPEG format.
- A configurable heartbeat to the RabbitMQ C++ and Java client transports is now available.
- The sniffer connector and adapter now supports copying packet payloads to an ESP blob field.
- You now can use the following as the message ID for hot failover:

- the RabbitMQ delivery tag
- the Kafka offset

For more information, see *SAS Event Stream Processing: Connectors and Adapters*.

Enhancements to ESPPy

ESPPy now enables you to group windows that are connected in a specific way into templates, which then can be reused within a Python program. For more information, see "Creating and Using Templates" in *SAS Event Stream Processing: Using the Python Interface*.

ESPPy now provides the capability to visualize ESP models in action within a Jupyter Lab environment. You can create various charts, an ESP model viewer, and an ESP log viewer. For more information, see "Visualization Programming Objects in ESPPy" in *SAS Event Stream Processing: Using the Python Interface*.

New ESP Server Connection API

A new API that uses the WebSocket protocol is available that enables ESP functions to operate with minimal system resources. The API sets up a single WebSocket connection to handle all ESP server activities. You can have a single WebSocket connection that does the following actions:

- subscribes to any number of ESP windows
- publishes events
- receives server status information
- monitors the server log

For more information, see "Using the ESP Server Connection API" in *SAS Event Stream Processing: WebSocket API Reference*.

New Capabilities for the RESTful API

The ESP server now supports HTTP requests over Kafka and MQTT message queues, processing those requests as if they were sent over a TCP/IP socket. For more information, see "Using the RESTful API over Message Queues" in *SAS Event Stream Processing: Using the ESP Server*.

New Configuration Capabilities

The configuration file **`esp-properties.yml`** has been expanded to include two new sections:

- The `env` section enables you to set additional environment variables.
- The `prometheus` section enables you to expose ESP server metrics to Prometheus, an open-source application used to monitor events and alerts in a cloud-based environment.

For more information, see "Configuring the ESP Server" in *SAS Event Stream Processing: Using the ESP Server*.

THE POWER TO KNOW.