Reference architecture

SAS® Viya™ edge-to-enterprise IoT analytics

The number of devices connected to the internet is growing exponentially, with Gartner projecting 8.4 billion in use this year.[1] These devices and the connections among them are known as the Internet of Things (IoT). Every one of these billions of devices and the sensors attached to them produces data. The volume of information is enormous, as is its potential to provide businesses with actionable insights. But businesses can realize this potential only when they have effective tools to gather, analyze, and control the massive amounts of IoT data available to them.

In one popular model, systems collect data from IoT devices and sensors in remote—or edge—locations and transfer this information to the datacenter, where it remains stored until businesses get around to analyzing it. While this approach is viable in certain industries and when a limited number of devices are generating a small volume of data, it becomes less effective at scale and in settings where timing is critical. Many use cases warrant a method where the data from real-time sensors undergoes real-time analysis, enabling real-time action.

The SAS® Viya™ edge-to-enterprise IoT analytics platform moves data analysis to the edge. This can reduce the time to gain actionable insights from the data. It can also reduce the footprint of the data through intelligent data-reduction techniques, which has great value in scenarios that require transporting data from the collection points to the datacenter or cloud using cellular technology.

The core component of the solution is the SAS Event Stream Processing (SAS Viya Enabled) product that supports real-time analytics on data in motion. SAS ESP has multiple flavors that can run on a wide range of hardware in both the edge and datacenter. These instances of SAS ESP can allow businesses to gather and process data streams at the edge, then transfer them to the datacenter for further processing. The SAS IoT analytics platform also includes a suite of supporting software applications to help users analyze and mine IoT data and develop analytical models that assist with real-time analytics and processing.

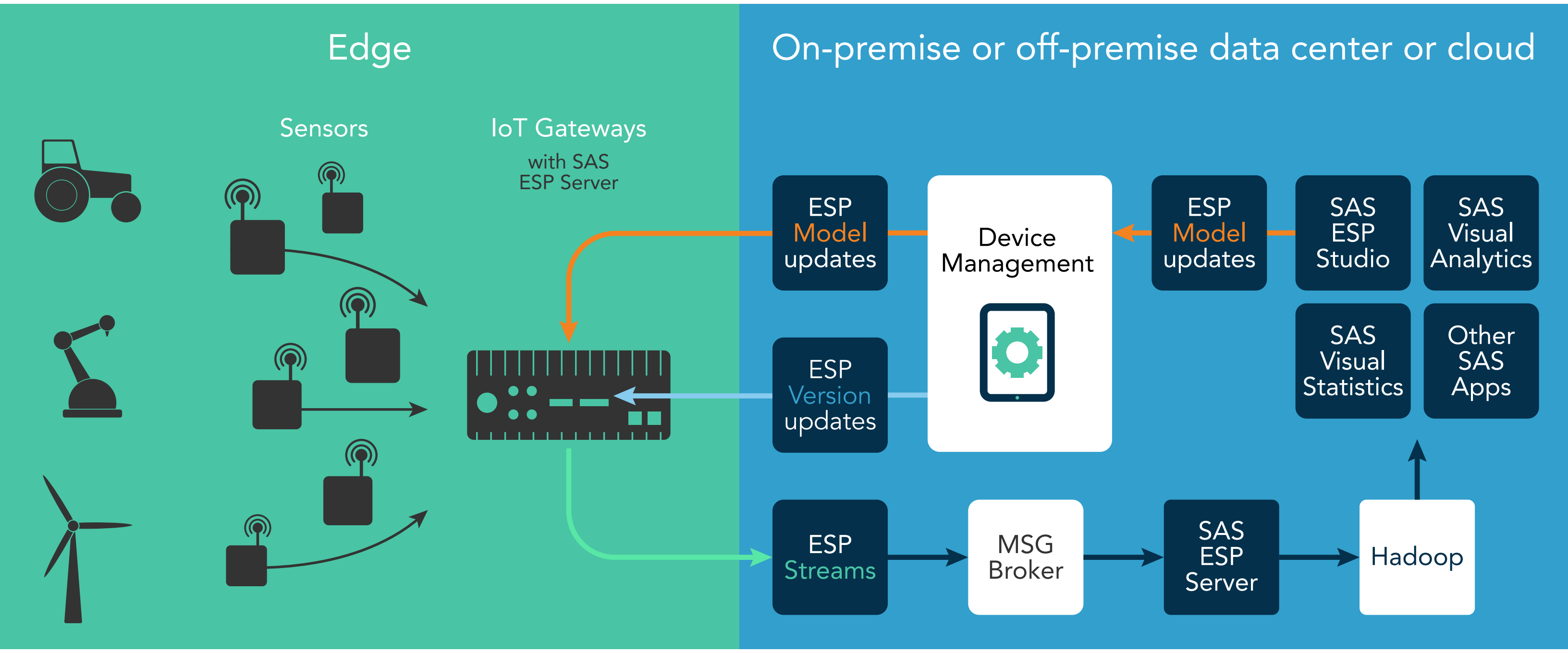
About the SAS® Viya™ IoT analytics platform

Software components

The SAS Viya IoT analytics platform includes the following software components:

- SAS Event Stream Processing (SAS Viya Enabled)
  - SAS Event Stream Processing for Edge Computing
  - SAS Event Stream Processing for Central Analytics Server
  - SAS Event Stream Processing Analytics
  - SAS Event Stream Manager
  - SAS Event Streamviewer
  - SAS Event Stream Studio
- SAS Visual Analytics (on SAS Viya): includes both SAS Visual Analytics and SAS Visual Statistics

The diagram below shows the SAS Viya platform and the logical flow of data from one component to the next in an edge-to-datacenter environment as outlined in SAS documentation.



SAS® Event Stream Processing for Edge Computing

By deploying SAS ESP for Edge Computing in the field, businesses can leverage devices that have limited storage and compute power to perform real-time analytics on data at the edge. This can help with large-scale deployments, allowing individual locations and gateways to process data before forwarding it to a central location for further processing and analysis.

SAS® Event Stream Processing, SAS® ESP Studio, and SAS® ESP Streamviewer

SAS ESP aims to help businesses make sense of the streaming data that flows in from daily operations, payment transactions, sensors, and any other IoT devices. By processing and running analytics on data in motion, businesses can potentially reduce the time to gain actionable insights. For example, SAS ESP can support a range of analytics on data in motion, from data management to predictive analytics to machine learning. Once SAS ESP has processed the data in motion, it can move to the big data environment for performing heavy-duty analytics with SAS Visual Analytics, SAS Visual Statistics, and other components of SAS Viya.

SAS ESP includes a portfolio of available development, visualization, and management tools such as SAS ESP Studio, a web-based user interface that can aid administrators and analysts. SAS ESP Studio allows analysts to develop data flow-driven analytical models and supports both a drag-and-drop interface and a developer-friendly XML interface. By visualizing ESP processing models, users can see how data flows work and interact with one another. SAS ESP Streamviewer enables users to subscribe to event streams, view tables and graphs in real time, and customize and save dashboards so they can access the same views of data in motion at a later time.

SAS® Visual Analytics and SAS® Visual Statistics

SAS Visual Analytics supports visualization of massive volumes of IoT data. It allows analysts to create robust, interactive reports that reveal insights related to asset performance and operational efficiency, along with providing predictive analysis (see Figure 1). Analysts can also use SAS Visual Analytics to integrate sensor data with location data, one of the great benefits of the IoT. With edge devices situated in different locations around the company and sensors located on valuable hardware off site, businesses can visualize sensor and location data simultaneously. With SAS Visual Statistics, users can apply advanced analytical techniques to analyze the IoT data and develop analytical models to keep up with the vast and changing IoT landscape.

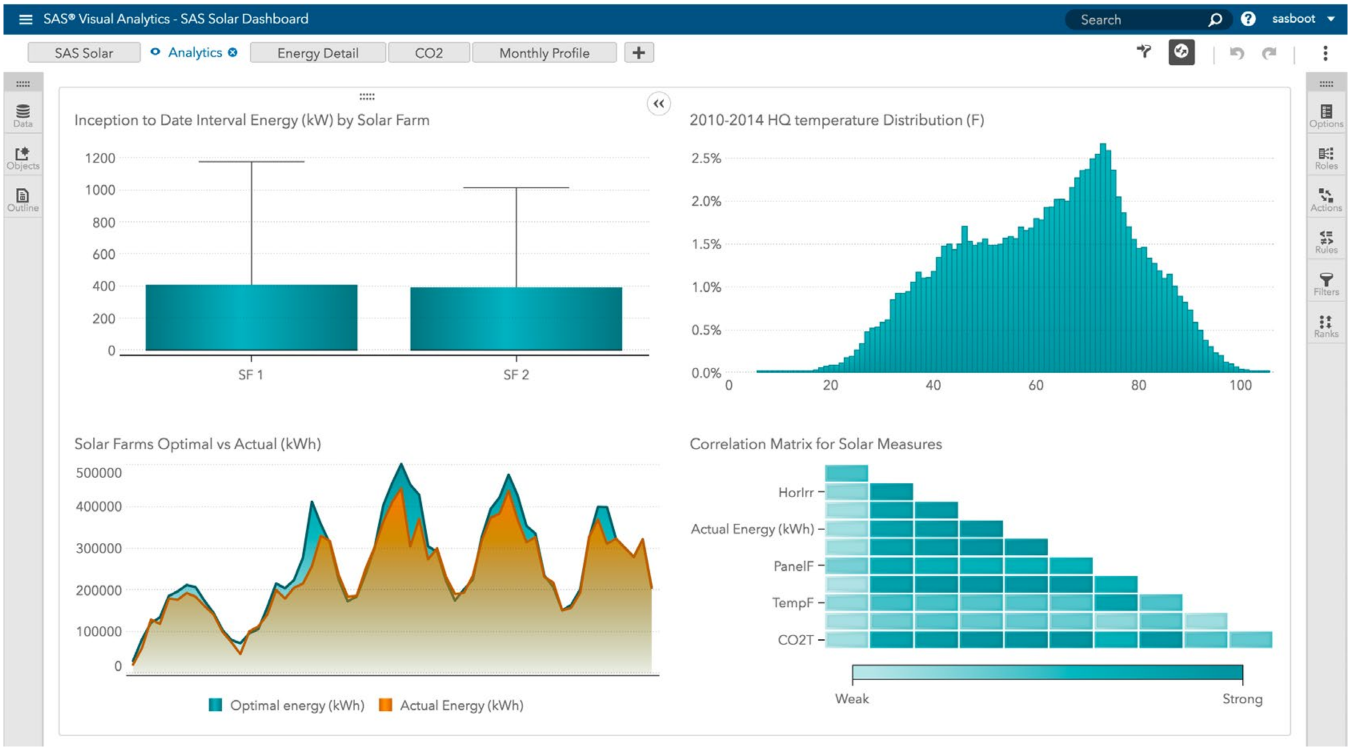


Figure 1: A SAS Visual Analytics report showing correlation variables.

Hardware recommendations

SAS® ESP for Edge devices

SAS ESP for Edge Computing includes multiple types of connectors and adapters that SAS ESP programmers and developers can use to create SAS ESP data models, flows, and processing schemes. These connectors and adapters can stream data into or out of SAS ESP and can interface with many communication fabrics, drivers, and clients. The table accessible through the link below shows the connectors and adapters available with SAS ESP for Edge Computing. In addition, different ESP engines can also stream data to one another, in the form of ESP-ESP connectors and adapters.

SAS documentation breaks down these connectors and adapters in detail. More information is available at:

http://documentation.sas.com/?cdcId=espcdc&cdcVersion=4.3&docsetId=espca&docsetTarget=p1swscq8yglnunn1p44y46w2rjtx.htm&locale=en

SAS ESP for Edge Computing system requirements are minimal, and can run on even devices with as little as 1 GB of RAM.

Hardware recommendations for SAS ESP for Edge Computing

Item	Recommended level*
CPU	Dual or quad core x86_64 compatible processor
Memory	4 GB of available RAM
Disk space	1 GB or more of free space for the installation

* The bare minimum requirements for an installation of SAS Event Stream Processing for Edge Computing are a single-core x86_64 compatible processor, 1 GB of available RAM, and 700 MB of free disk space. Source: http://documentation.sas.com/api/docsets/dplyedge0phy0lax/4.3/content/dplyedge0phy0lax.pdf#nameddest=titlepage

SAS® ESP

Deploying SAS ESP in the datacenter on more powerful hardware than is available at the edge enables SAS ESP to handle a larger volume of data, such as that from multiple edge devices and other SAS ESP installations.

The hardware requirements for the datacenter version of SAS ESP are greater than for the SAS ESP for Edge Computing variant, but SAS ESP can still run on commodity hardware as well as on many cloud providers.

Hardware recommendations for SAS ESP

Item	Recommended level*
CPU	4 cores (x86 architecture) Intel® Xeon® chip set with a minimum speed of 2.6GHz
Memory	8-16 GB of RAM Memory clock speed of 1600MHz
Disk space and speed	10 GB 10,000 RPM

*The bare minimum requirements for an installation of SAS Event Stream Processing are 4 cores, 4 GB of memory, and 1 GB of disk space. Source: http://documentation.sas.com/api/docsets/dplyesp0phy0lax/4.3/content/dplyesp0phy0lax.pdf#nameddest=titlepage

SAS® Visual Analytics and SAS® Visual Statistics

SAS recommends deploying SAS Visual Analytics and SAS Visual Statistics using a popular Python tool called Ansible. Once you have edited a few files, Ansible can automate the deployment of SAS Visual Analytics and SAS Visual Statistics on a single host or a large-scale distributed environment.

For more information on Ansible, see https://www.ansible.com/how-ansible-works.

After SAS ESP processes IoT data in real time, the data comes to rest in one of many storage options that SAS supports. From that point, the deep mining and advanced analytics available with SAS Visual Analytics and SAS Visual Statistics provide for reporting and discovery.

SAS Visual Analytics and SAS Visual Statistics can run on commodity hardware, and SAS also supports them on many cloud platforms. The table below shows the system requirements for SAS Visual Analytics and SAS Visual Statistics.

Requirements for single-machine deployment, full (all user interfaces)

Item	Minimum level
CPU	Intel Xeon CPU with 4 cores x86 architecture with a minimum speed of 2.6GHz
Memory	64 GB of RAM Memory clock speed of 1600MHz
Disk space and speed	2 x 300GB 10,000 RPM

Requirements for single-machine deployment, programming interface only

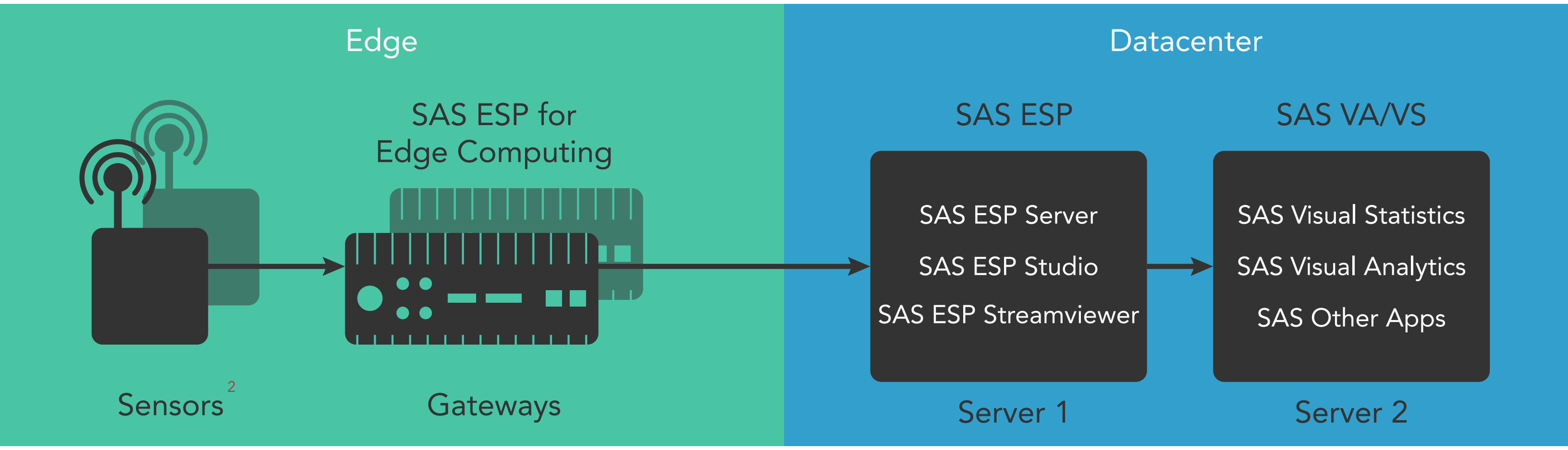
Item	Minimum level
CPU	Intel Xeon CPU with 4 cores x86 architecture with a minimum speed of 2.6GHz
Memory	32 GB of RAM Memory clock speed of 1600MHz
Disk space and speed	2 x 300GB 10,000 RPM

Reference architecture: Principled Technologies implementation

Now that you have some understanding of the SAS Viya components and how they can work together to help your business harness the potential of real-time data analysis, let's turn to the approach we used in our implementation. We used the following SAS Viya components and versions.

- SAS Event Stream Processing 4.3 for Edge Computing
- SAS Event Stream Processing 4.3
- SAS Event Stream Studio
- SAS Event Streamviewer
- SAS Visual Analytics 8.1
- SAS Visual Statistics 8.1

Below, we present a logical diagram of our test configuration and the SAS Viya components we implemented.



Hardware components

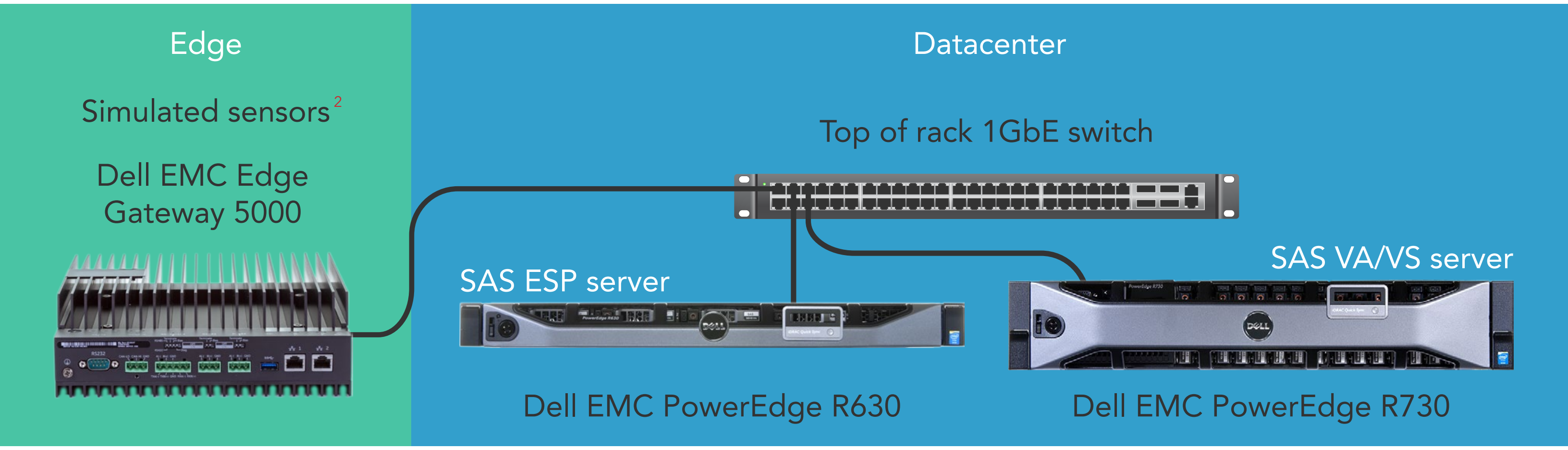
This SAS Viya implementation for edge-to-enterprise IoT analytics relies in part on the availability of compute resources throughout the network and the ability to leverage those resources for distributed processing. In the solution we tested, Intel Xeon processors provide those compute resources–but that's not all Intel has to offer for big data analytics.

Intel has made significant investments in analytics and the internet of things. Intel Xeon and Atom series processors power multiple solutions for end-to-end computing in the fog and at the edge.[3,4] At the edge, Intel also supports a number of operating systems and several manageability options from the Wind River Helix portfolio.[5] This portfolio supports an analytics environment with "a layered security approach," which can make Intel a strong choice for enabling an edge-to-enterprise analytics platform.[6]

Principled Technologies deployed SAS software onto three different dedicated systems, all powered by Intel processors and each with its own role. SAS ESP, SAS ESP Streamviewer, and SAS ESP Studio all ran on a Dell EMC™ PowerEdge™ R630 in the PT datacenter. SAS Visual Analytics and supporting software ran on a Dell EMC PowerEdge R730, also in the datacenter. SAS ESP for Edge Computing ran on a Dell Edge Gateway 5000, a dedicated, low power device in a different location in the same building. We connected all three systems to the same 1GbE network and subnet. SAS supports many other configurations that can increase fault tolerance and storage capacity and leverage multiple servers for increased compute power.

We connected all three systems to the same 1GbE network and subnet. SAS supports many other configurations that can increase fault tolerance and storage capacity and leverage multiple servers for increased compute power.

Below, we present a physical diagram of our test configuration and the hardware components we implemented.



Gateway

In our configuration, the Dell Edge Gateway 5000 ran the Wind River® operating system[7] and included 2 GB of DDR3 RAM and 32 GB of SSD storage. It is available with cellular LTE support and provides connectivity options for many IoT devices, including multiple serial connections, CANbus, and available ZigBee mesh networking capability.

Figure 2: The Dell Edge Gateway 5000.

For more information on the Dell Edge Gateway 5000, see http://www.dell.com/en-us/work/shop/productdetails/dell-edge-gateway-5000/xctoi5000us

More information on Wind River Linux® and the Dell Edge Gateway 5000 is available here: http://www.dell.com/support/manuals/us/en/19/dell-edge-gateway-5000/dell-edge-gateway-5000_Users_Guide/Wind-River-Linux?guid=GUID-6AE02B38-1784-434B-A999-3447B6133E1C&lang=en-us

SAS ESP server node

Companies can deploy SAS ESP on a commodity server or in the cloud. In our test installation, we put all the components of SAS ESP on a single Dell EMC PowerEdge R630. You can spread the components of SAS ESP across separate machines to meet the needs of your business. For instance, you can install SAS ESP Streamviewer on a separate machine than SAS ESP Studio.

SAS ESP	
Model	Dell EMC PowerEdge R630
CPU	2 x Intel Xeon processor E5-2603 v3
RAM	64GB DDR4
Disk(s)	2 x 300GB 10K 6Gb SAS

SAS® Visual Analytics/SAS® Visual Statistics server node

Companies can deploy SAS Viya in many configurations to fit their needs. In our test deployment, we deployed all components of SAS Visual Analytics and SAS Visual Statistics on a single Dell EMC PowerEdge R730. Other types of deployments may include multiple servers for different components of SAS Viya for a high-performance distributed environment. Companies can also deploy components of SAS Viya on many cloud providers.

SAS VA	
Model	Dell EMC PowerEdge R730
CPU	2 x Intel Xeon Processor E5-2690 v3
RAM	128GB DDR4
Disk(s)	2 x 300GB 10K 6Gb SAS

Deploying SAS® Viya™ IoT Analytics

While you can deploy the three pieces of hardware in this validation in any order, we started with the gateway device, then deployed SAS ESP in the datacenter, and lastly deployed SAS VA/VS, also in the datacenter.

Downloading and deploying SAS® ESP for Edge Computing

To install SAS ESP on the Dell Edge Gateway 5000 running Wind River Linux 7.0.0.19, Intelligent Device Platform XT 3.1 / LuCI (0.12) requires a Linux jump server with the ability to create a local yum repository. This installation method does not use the deployment script from the SAS order. We used a virtual machine running Red Hat Enterprise Linux 6.7 to create the local yum repository necessary for installation.

An order email from SAS should include four files:

- **SAS_CA_Certificate.pem**: This file establishes the SAS certificate authority.
- **entitlement_certificate.pem**: This file uses the established authority to download files from the repository.
- **SASViyaV0300_09KJH4_Linux_x86-64.txt**: The license file.
- **ESP_Edge_Script.sh**: Use this customized script in your software deployment. The deployment guide will describe where in the process to use the script. Do not use this file for Yocto/Wind River Linux deployment.

1. SSH or login to the jump/prep server.
2. Copy the three files (the certificates and the license file) to the jump/prep server, and to a directory of your choice.
3. Run the following command:
   ```
   sudo yum install yum-utils createrepo
   ```
4. Set the environment variable for the certificates. The following command sets it to the working directory:
   ```
   export CERTDIR=$(pwd)
   ```
5. Download the repo RPM using CURL:
   ```
   curl -OLv --cert $CERTDIR/entitlement_certificate.pem --cacert $CERTDIR/SAS_CA_
   Certificate.pem https://ses.sas.download/ses/repos/meta-repo//sas-meta-repo-1-1.
   noarch.rpm
   ```
6. Install the downloaded RPM file using the following command:
   ```
   sudo yum install sas-meta-repo-1-1.noarch.rpm
   ```
7. Copy the entitlement certificate to /etc/pki/sas/private
   ```
   cp $CERTDIR/entitlement_certificate.pem /etc/pki/sas/private/entitlement_
   certificate.pem
   ```
8. Install the software using yum:
   ```
   yum install sas-espedge-100-x64_yocto_linux_1-yum
   ```
9. When prompted, type `y` to install.
10. Create the local repository:
    ```
    reposync -r sas-espedge-100.4.0-x64_yocto_linux_1-yum
    ```
11. Log into the Edge device, and use the following command to create a directory on which to copy the .rpm files:
    ```
    mkdir espedge
    ```
12. Log into the remote repository machine, and use the following command to copy the files to the Edge device:
    ```
    scp -r sas-espedge-100.0.0-x64_yocto_linux_1-yum root@10.41.2.72:/opt/espedge
    ```
13. From the Edge device, use the following command to unpack and install the .rpm files:
    ```
    rpm -Uvh /opt/espedge/sas-espedge-100-x64_yocto_linux_1-yum/*.rpm
    ```
14. Copy the license file to the Edge device from the remote repository machine using the following command:
    ```
    scp -r SASViyaV0300_9BRGKW_Linux_x86-64.txt root@10.41.2.72:/opt/sas/viya/home/
    SASEventStreamProcessingEngine/4.3.0/etc/license
    ```
15. Set the environment variables on the edge device to prepare to start SAS ESP for Edge Computing:
    ```
    export DFESP_HOME=/opt/sas/viya/home/SASEventStreamProcessingEngine/4.3.0
    export LD_LIBRARY_PATH=$DFESP_HOME/lib:/opt/sas/viya/home/SASFoundation/sasexe
    ```
16. Wind River Linux doesn't include libnuma/numactl out of the box, which is necessary for SAS ESP. Download the numactl source from Github on the edge device:
    ```
    git clone https://github.com/numactl/numactl.git
    ```

17. Navigate into the downloaded directory:
```
cd numactl
```
18. Run autogen:
```
./autogen
```
19. Run configure:
```
./configure
```
20. Run make:
```
make
```
21. Install the software:
```
make install
```
22. Create a symlink to the libnuma shared object in the SAS ESP Library folder:
```
ln -s /usr/local/lib/libnuma.so.1 /opt/sas/viya/home/SASFoundation/sasexe/
```

Installing Red Hat Enterprise Linux 6.7

Perform the following steps on both the Dell EMC PowerEdge R630 and the Dell EMC PowerEdge R730.

1. Press the Power button.
2. During boot, press F11 to enter the Boot Menu.
3. Select One-Shot Boot.
4. Select the USB or other installation media.
5. Wait for the server to boot into the RHEL installer.
6. Select Install Red Hat Enterprise Linux, and press Enter.
7. Select the proper language, and press Enter.
8. Select the proper keyboard type, and press Enter.
9. Select the type of media where the installation files are located, and press Enter.
10. Select the installation directory, and press Enter.
    A graphical user interface opens.
11. On the RHEL splash screen, click Next.
12. Leave the default device options, and click Next.
13. Click Yes, discard any data.
14. Provide a Hostname for the server, and click Next.
15. Provide the correct time zone, and click Next.
16. Provide a root password, and click Next.
17. Select Use all Space, and click Next.
18. Select the drive on which to install Red Hat Enterprise Linux, and press the right arrow.
19. Select the drive now in the right pane, drag the bootloader onto that drive, and click Next.
20. Click Write changes to disk.
21. Leave the default installation settings, and click Next.
    Red Hat Enterprise Linux should install.
22. Click Reboot.

Updating Red Hat Enterprise Linux

Perform the following steps on both the Dell EMC PowerEdge R630 and the Dell EMC PowerEdge R730.

1. Log into the machine via SSH or locally.
2. Run the following command, and enter the correct credentials:
   `subscription-manager register`
3. Run the following command:
   `subscription-manager attach`
4. Run the following command:
   `subscription-manager release –set=6.7`
5. Run the following command, and let all updates install:
   `yum update –y`

Downloading and installing SAS® ESP

1. Log into the Dell EMC PowerEdge R630 via SSH.
2. Download or transfer the four files from the SAS order to the server hosting SAS ESP (the Dell EMC PowerEdge R630).
3. Set the following environment variables in the shell using the following commands:
   ```
   export DFESP_HOME=/opt/sas/viya/home/SASEventStreamProcessingEngine/4.3.0
   export LD_LIBRARY_PATH=$DFESP_HOME/lib:/opt/sas/viya/home/SASFoundation/sasexe
   export PATH=$PATH:$DFESP_HOME/bin
   ```
4. Change the working directory to the location with the four files from the SAS order.
5. Run the deployment script:
   `./customized_deployment_script.sh`
6. To install sas-meta-repo, type y
7. To install sas-esp-101-x64-redhat_linux_6-yum, type y
8. To install sas-certframe, sas-espconbase, sas-espcondb, sas-espconext, sas-espcxxdev, sas-espexam, and their dependencies, type `y`
9. To accept the certificate, type `y`
10. To install the SAS ESP Streamviewer (sas-espstrmvwr), type `y`
11. To install the SAS ESP Event Stream Manager Agent, type `y`
12. To install the SAS Text Analytics Component (sas-txtmineng), type `y`
13. To install the SAS Event Stream Processing Studio (sas-espvm), type `y`

Configuring SAS ESP

1. SAS ESP requires Java 1.8.0. Check the Java version with the following command:
   `java –version`
2. Red Hat Enterprise Linux 6.7 ships with openjdk 1.7.0. Uninstall this with the following command:
   `yum remove java-1.7.0-openjdk`
3. Download the .rpm Java installer from Oracle: http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
4. Install the Java .rpm with the following command:
   `rpm -ivh jdk-8u131-linux-x64.rpm`

5. Re-check the Java version to ensure it is now 1.8.0.
6. Start the SAS Event Stream Processing Studio with the following command:
```
service sas-viya-espvm-default start
```
7. Configure the Red Hat Enterprise Linux 6.7 firewall to allow traffic on port 8080 (the default port for SAS ESP Studio). The following command provides a simple wizard:
```
system-config-firewall-tui
```
The SAS ESP Studio should now be available at the following URL: http://esp-studio-hostname:port/SASEventStreamProcessingStudio
8. Start the Event Stream Processing XML Server with the following command:
```
./opt/sas/viya/home/SASEventStreamProcessingEngine/4.3.0/bi/dfesp_xml_server
-pubsub n -http-admin adminport -http-pubsub pubsubport &
```
9. Navigate to the SAS ESP Studio URL in a web browser.
10. Click Projects.
11. Click Add.
12. In the dialog that appears, complete the fields. (Note: Starting SAS ESP XML Server configures the ports.)
13. Click OK.

Deploying SAS® Visual Analytics

Installing prerequisites

1. Transfer the contents from the SAS order email to the server, into a directory such as /sas/install. Be sure to configure the hostname properly. All versions of the hostname must resolve and be ping-able. The hostname must have both a long and short version, and all the following commands must output the correct name:
```
hostname
hostname -f
hostname -s
hostname -A
hostname -a
```
2. Enable the repository rhel-6-server-optional-rpms with the following command:
```
subscription-manager repos --enable rhel-6-server-optional-rpms
```
3. Install EPEL (Extra Packages for Enterprise Linux) with the following command:
```
rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
```
4. Configure SELinux with the following commands:
```
setenforce 0
sed -i.bak -e 's/SELINUX=enforcing/SELINUX=permissive/g' /etc/selinux/config
```
5. To make sure that /root/.ssh/known_hosts exists, use SSH to log into itself (localhost) or another server. Note: This is only valid if the server has never used SSH before.
6. To configure the firewall properly, use the Red Hat Enterprise Linux tool available with the following CLI command:
```
system-config-firewall-tui
```

7. Install Ansible, a Python utility that aids in the installation of SAS Viya, with the following command:

```
sudo yum install -y gcc automake openssl-devel python-devel libffi-devel
sudo yum install -y python-crypto python-paramiko python-keyczar python-setuptools python-pip python-six
sudo yum install -y python-virtualenv
mkdir virtwork
cd virtwork
virtualenv deployment
source deployment/bin/activate
pip install ansible==2.2.1
```

8. You are now inside a Python virtual environment which has Ansible installed. To return to the Python virtual environment, use the following command:

```
source <dir>/deployment/bin/activate
```

Creating a SAS® installer user account

1. SSH to the Dell EMC PowerEdge R730, and enter the appropriate root credentials.
2. Create the user account to install SAS with the following command:

```
useradd cas
```

3. Set the password for the new account with the following command:

```
passwd cas
```

4. Create a group named sas with the following command:

```
groupadd sas
```

5. Ensure the user cas has sudo access. Edit the sudoers file by running the following command:

```
visudo
```

and removing the comment on the following line:

```
%wheel        ALL=(ALL)       ALL
```

6. Add the cas user to the wheel group with the following command:

```
usermod -aG wheel cas
```

Changing ulimits as recommended by SAS

1. Log into the system as the root user account.
2. Edit the /etc/security/limits.conf file, and add the following lines:

```
@sas              soft    nofile          20480
@sas              hard    nofile          63536
```

3. Edit the /etc/security/limits.d/90-nproc.conf file, and add the following lines:

```
@sas              soft    nproc         65536
@sas              hard    nproc         65536
```

4. Verify settings by using the following command:

```
ulimit -a
```

Installing SAS® Viya™ using Ansible

1. Enter the previously created deployment Python virtualenv using the following command:
```
source <dir>/deployment/bin/activate
```
2. Navigate to the directory where the files from the SAS order email reside.
3. Extract the SAS_Viya_playbook.tgz file with the following command:
```
tar xf SAS_Viya_playbook.tgz
```
4. Edit the vars.yml file with the following command:
```
nano vars.yml
```
5. Edit any necessary parts of the vars.yml file as needed for the specific deployment. For our deployment, we only changed the deployment label.
6. When deploying SAS Viya on a single host, run the following command:
```
ansible-playbook host_local site.yaml
```

Configuring and starting the validation workload

The validation workload consists of two SAS ESP model files and one comma-separated values (.CSV) data file. One model file goes on the edge device and the other is for the SAS ESP deployment in the datacenter. The edge device reads data from the .CSV file using a ESP-file adapter. Then, the SAS ESP for Edge Computing model performs data processing tasks, and sends the processed data to the SAS ESP installation in the datacenter using an ESP-ESP adapter. The SAS ESP installation in the datacenter includes SAS ESP Streamviewer, which lets users view this workload and streaming data in action.

SAS® ESP for Edge Computing

SAS provided the workload, data, and model files. They consist of a .CSV file and an .XML file that contained the SAS ESP model file. The model file defines certain processes and functions that SAS ESP performs on data.

1. SSH to the gateway device.
2. Copy the .XML model file and the .CSV data to the edge device.
3. Start SAS ESP for Edge Computing and specify the model file:
```
$DFESP_HOME/bin/dfesp_xml_server -http-admin 9900 -http-pubsub 9901 -pubsub 9902
–model file://<model file> xml &
```
4. Start the ESP-file adapter, and specify the data file:
```
$DFESP_HOME/bin/dfesp_fs_adapter -k pub -h dfESP://<edge hostname>:9902/PT_
Edge/Query_1/Solar_Farm_Data -t csv -f <file path and file> -b 1 -r 1 -x 1 -O -F
normal &
```

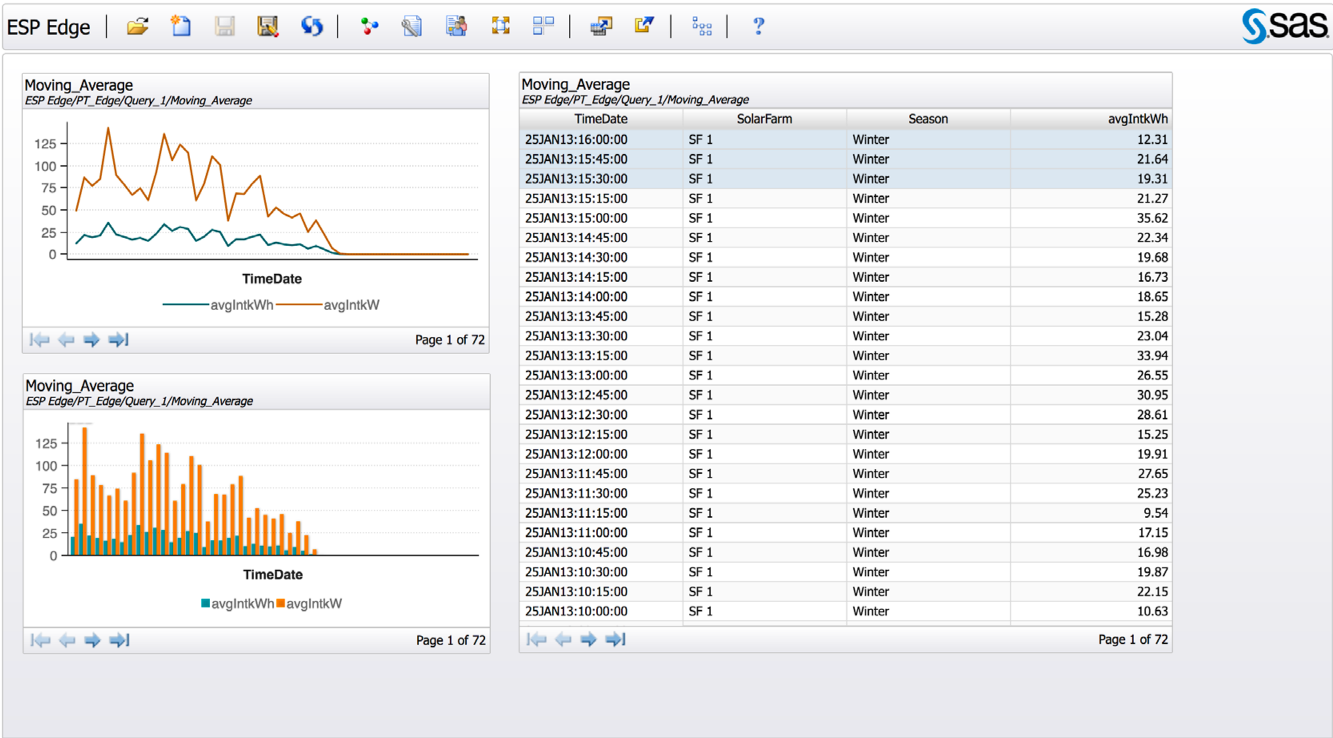


Figure 3: SAS ESP Streamviewer dashboard showing running average data for our test solar farm data.

SAS® ESP

1. SSH to the server.
2. Copy the .XML model file to the server hosting SAS ESP 4.3 (e.g. /home/sas/ or similar).
3. Ensure the correct environment variables are set:

```
export DFESP_HOME=/opt/sas/viya/home/SASEventStreamProcessingEngine/4.3.0
```

4. Start SAS ESP Server:

```
$DFESP_HOME/bin/dfesp_xml_server -http-admin 9950 -http-pubsub 9951 -pubsub 9952
–model file://<model file> &
```

5. Start the ESP-ESP adapter that connects the Edge installation of SAS ESP to the Datacenter installation of SAS ESP:

```
$DFESP_HOME/bin/dfesp_esp_adapter -s dfESP://<hostname-edge>:9902/PT_Edge/
Query_1/Moving_Average?snapshot=true?collapse=true -p dfESP://<hostname-
server>:9952/PT_Server/Query_1/Output_PT_Edge &
```

6. Start SAS ESP Streamviewer:

```
./streamviewer.sh –product h2 –http 5990 -h2file config.db
```

7. In a web browser, navigate to the IP address or DNS name of the system and the above port of SAS ESP Streamviewer.
8. In the dialog that appears, select New…
9. In the dialog that appears, provide a name for the SAS ESP Server, the hostname/IP address, and the port number. This is the port number of the http-pubsub from the above command when SAS ESP Server is started.
10. Click Test Connection… to verify settings. Click OK.
11. Repeat steps 8 through 10 for the SAS ESP for Edge Computing device.
12. Click Done.

Viewing a data model in SAS® ESP Streamviewer

1. In a web browser, navigate to the IP address or DNS name of the system and the port of SAS ESP Streamviewer.
2. Click Show Model.
3. Select the correct SAS ESP Server from the drop-down menu. Click Refresh.

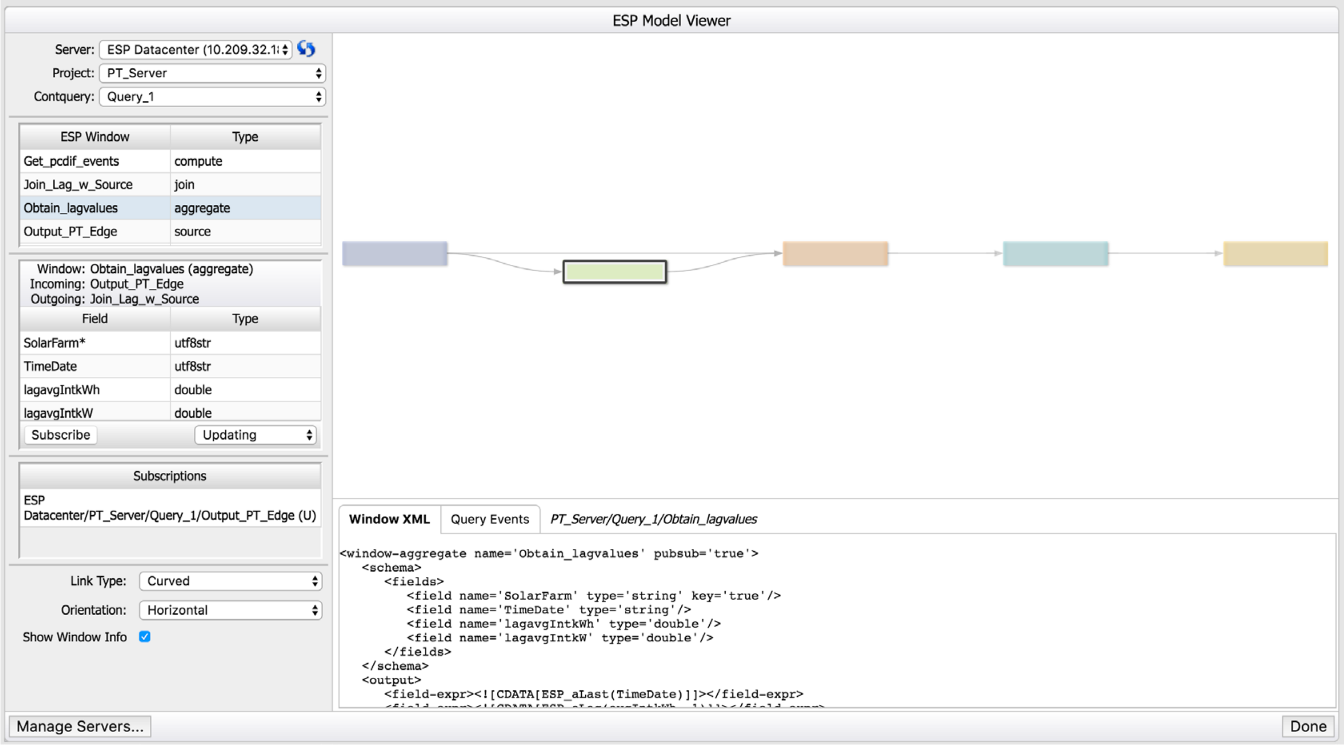


Figure 4: SAS ESP Model Viewer showing our test solar farm model.

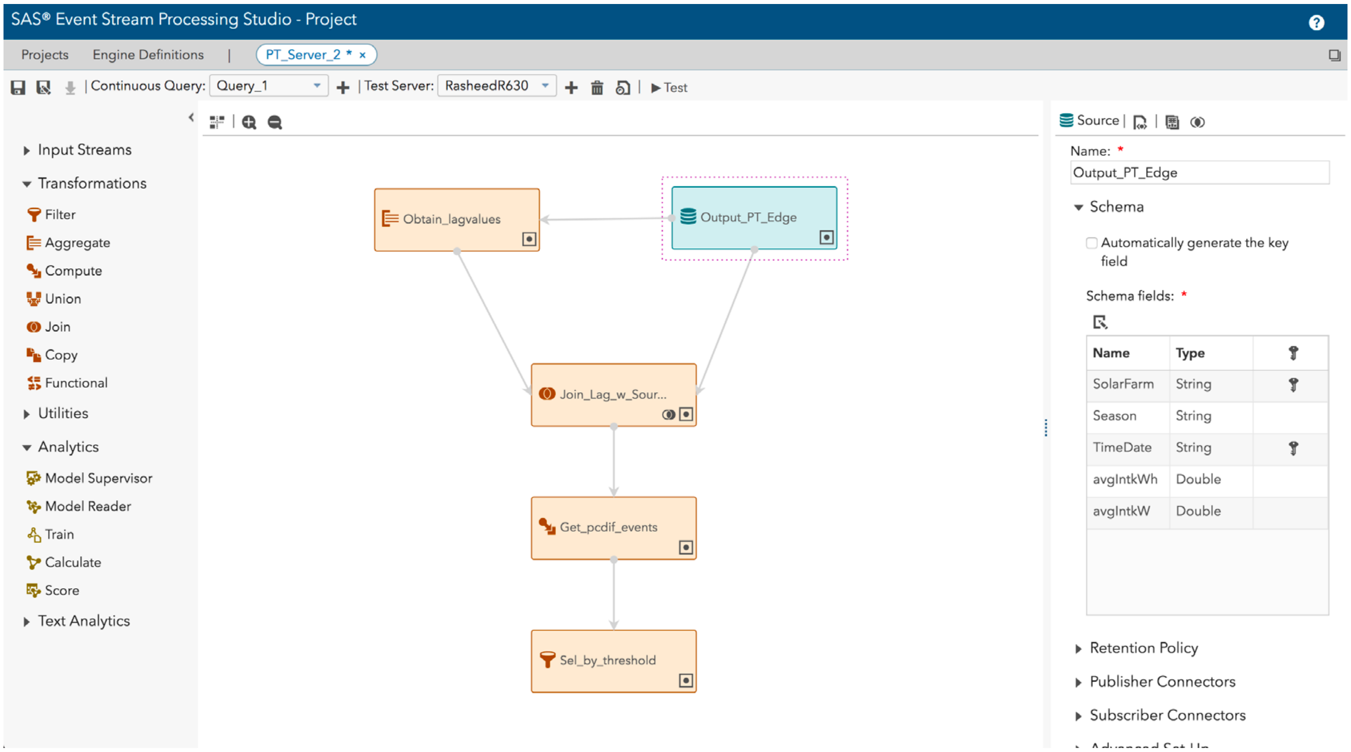
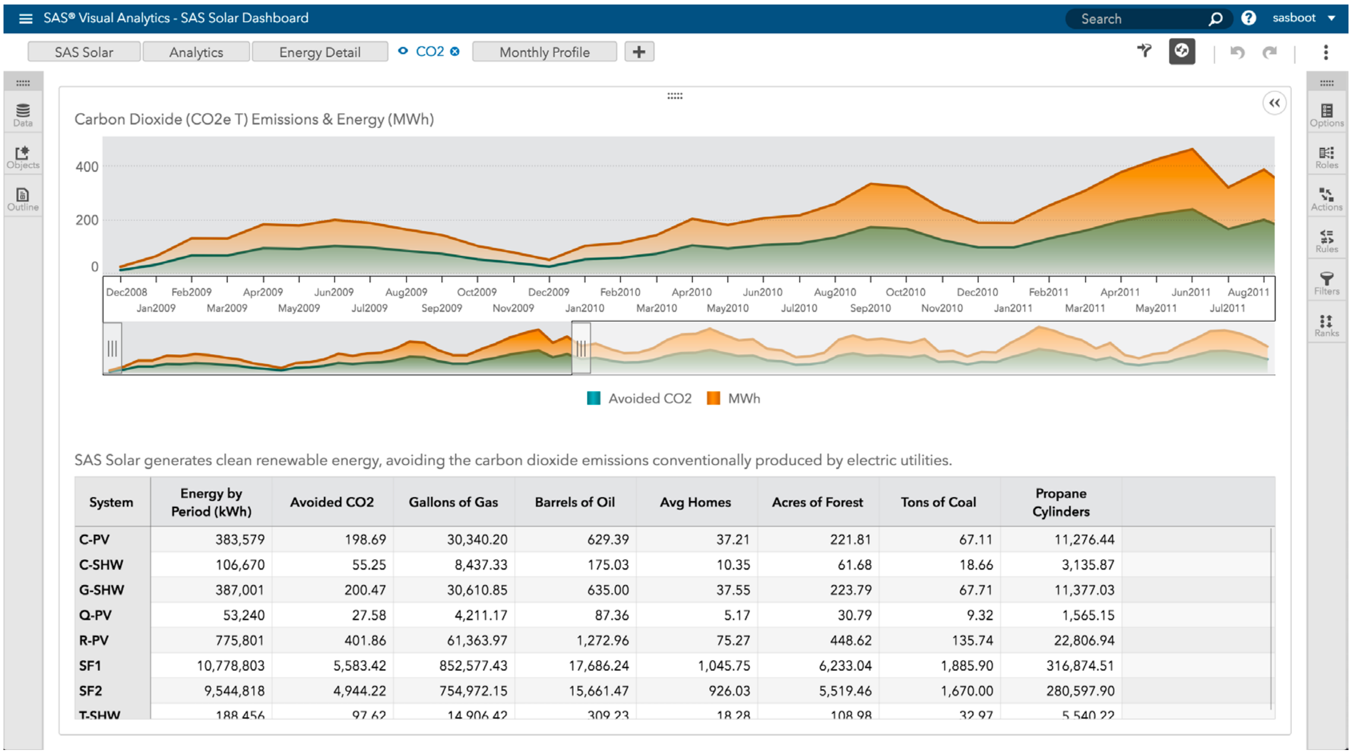


Figure 5: SAS ESP Studio showing our test solar farm project.

SAS® Visual Analytics Report

1. Using a web browser, navigate to SAS Home at http://<hostname or IP>/SASHome
2. Sign in using the appropriate credentials.
3. From the left menu, select SAS Visual Data Builder.
4. Select Open Table.
5. Select Import, and select SAS Data Set.
6. Select Choose Files, and select the data. Click OK.
7. From the left menu, select Reports.
8. Click Create a new Report.
9. To create a report from a previously saved XML file, press Control – Alt – B.
10. Select all the XML code in the window, and delete it. Paste the XML code from the proper source. Click Load.
    The report will load.
11. To save the report, click Save As, and provide a name. Click Save.



System	Energy by Period (kWh)	Avoided CO2	Gallons of Gas	Barrels of Oil	Avg Homes	Acres of Forest	Tons of Coal	Propane Cylinders
C-PV	383,579	198.69	30,340.20	629.39	37.21	221.81	67.11	11,276.44
C-SHW	106,670	55.25	8,437.33	175.03	10.35	61.68	18.66	3,135.87
G-SHW	387,001	200.47	30,610.85	635.00	37.55	223.79	67.71	11,377.03
Q-PV	53,240	27.58	4,211.17	87.36	5.17	30.79	9.32	1,565.15
R-PV	775,801	401.86	61,363.97	1,272.96	75.27	448.62	135.74	22,806.94
SF1	10,778,803	5,583.42	852,577.43	17,686.24	1,045.75	6,233.04	1,885.90	316,874.51
SF2	9,544,818	4,944.22	754,972.15	15,661.47	926.03	5,519.46	1,670.00	280,597.90
T-SHW	188,456	97.62	14,906.42	309.23	18.28	108.98	32.97	5,540.22

Figure 6: SAS Visual Analytics executive report showing avoided CO2 figures using our test solar farm data.

Conclusion

The Internet of Things has arrived, and the data is pouring in. From manufacturing to power production to fleet management, companies across many industries are experiencing an information explosion—in terms of both the types of data available to them and its sheer volume. One way to harness the power of this data is multiphase analytics, where a business can apply analytics at various points throughout the network and the datacenter and choose the correct type of analytics.

In this reference architecture, we have demonstrated that businesses could use the SAS Viya edge-to-enterprise solution to distribute analytics between the edge and the datacenter. Analytics running at the edge is closer to the data's source, and helps reduce latency in the decision-making process. Analytics that runs at the edge must work in harmony with datacenter and cloud analytics. Together, they establish a multiphase analytics lifecycle that allows organizations to quickly iterate as they reach new findings.

By providing access to real-time data from the edge and the tools that allow companies to visualize and analyze it, SAS Viya can help position companies to make the critical, timely business decisions that truly maximize the potential of IoT.

---

1 Gartner says 8.4 Billion Connected "Things" Will Be in Use in 2017, Up 31 Percent From 2016, accessed: 08/18/17, http://www.gartner.com/newsroom/id/3598917

2 In our testing, the IoT sensor data was provided by SAS

3 "Fog Computing Rolls In, Strengthens Smart Grid Security", accessed: 09/21/17, https://www.forbes.com/sites/inteliot-partnership/2016/11/01/fog-computing-rolls-in-strengthens-smart-grid-security/#f2d2515218ba

4 "The Intel IoT Platform", accessed: 09/21/17, https://www.intel.com/content/dam/www/public/us/en/documents/white-papers/iot-platform-reference-architecture-paper.pdf

5 See 4

6 See 4

7 Wind River is wholly owned by Intel. Learn more at https://www.windriver.com/

This project was commissioned by SAS.