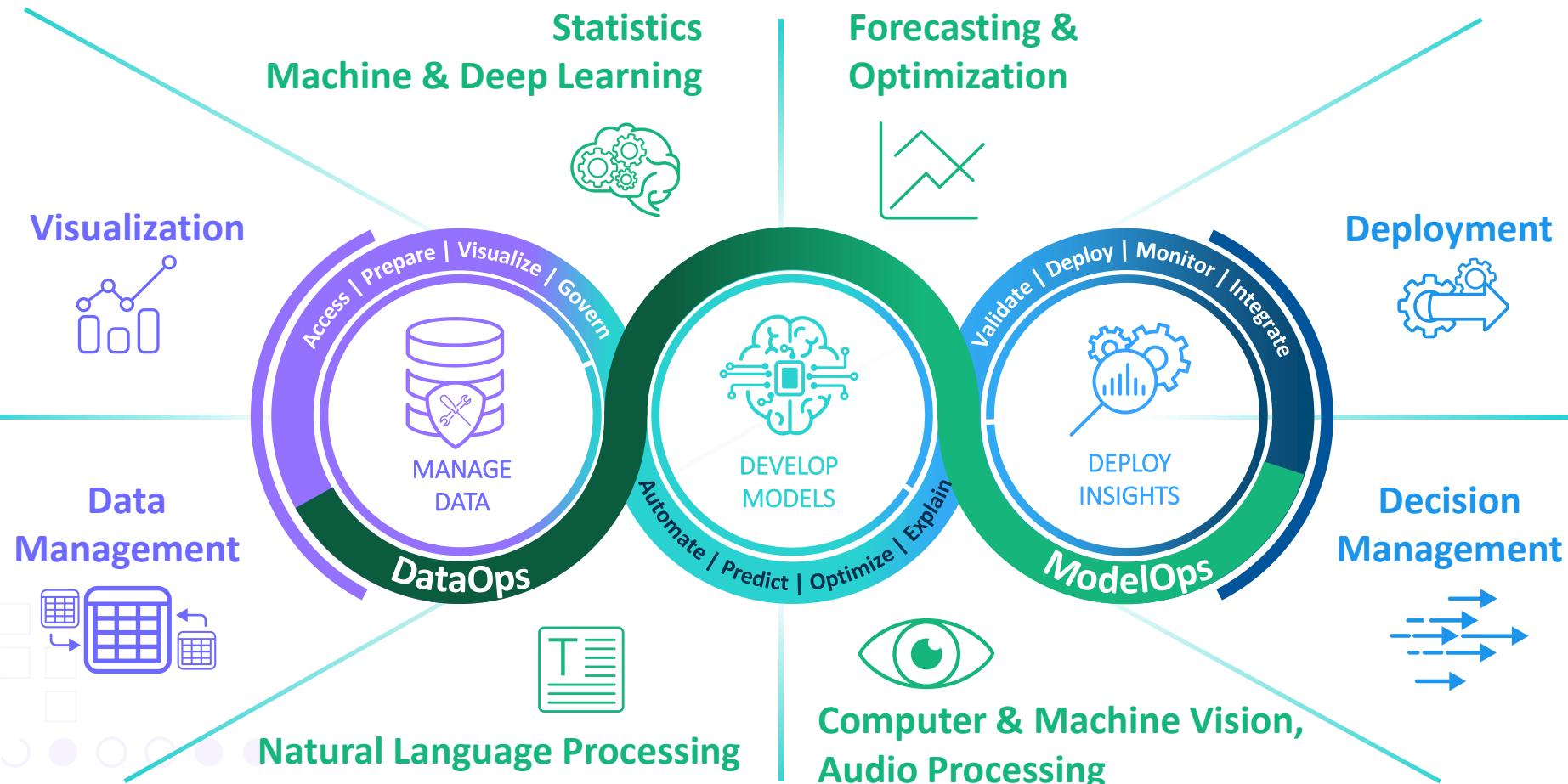


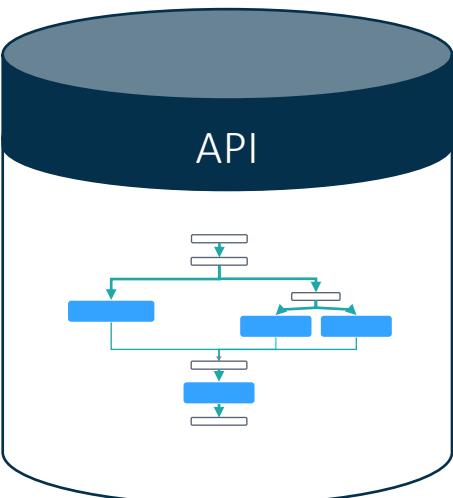
# SAS Plattform Netzwerktreffen 2022

SAS Viya - Deployment Optionen in der 'run time'

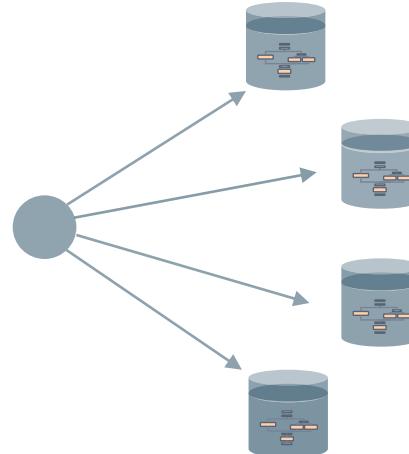
Phillip Manschek, SAS



# CONTAINERIZE



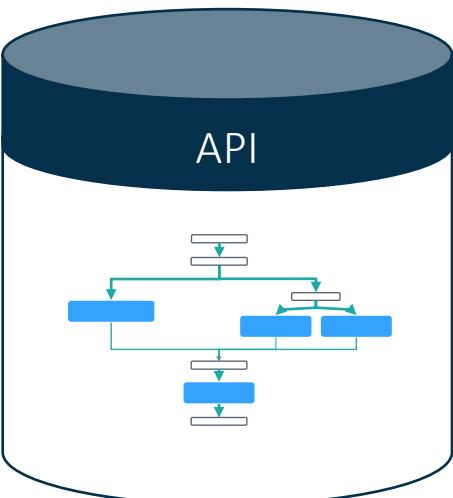
# DEPLOY



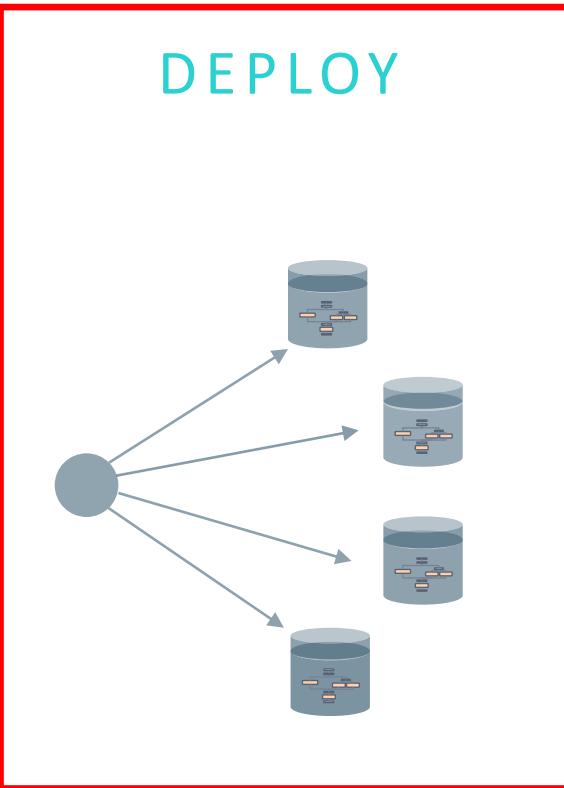
# INTEGRATE



## CONTAINERIZE



## DEPLOY



## INTEGRATE

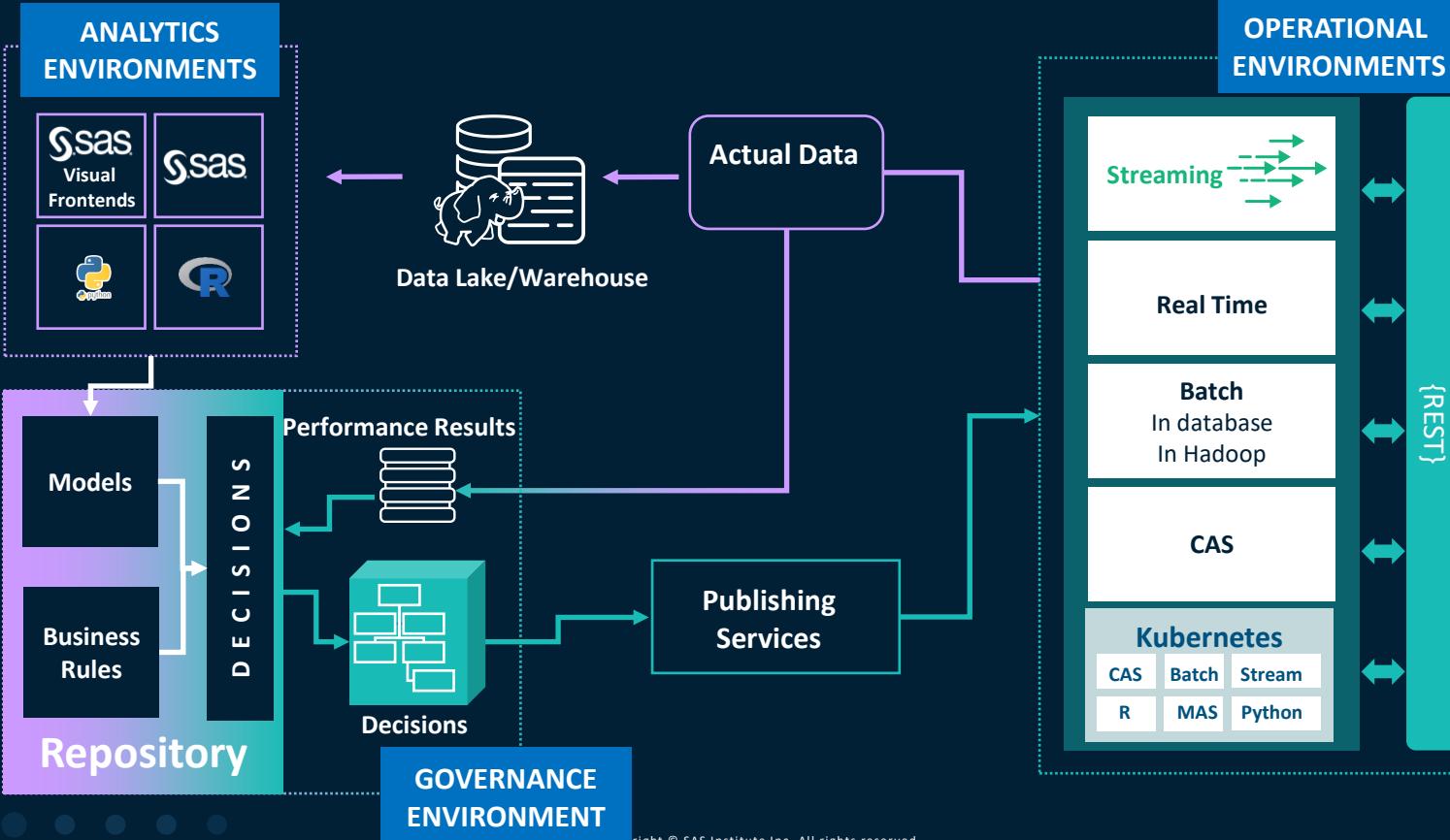


# Operationalizing Analytics

Dev & Register – Test – Deploy – Monitor - Improve



MODEL/DECISION PRODUCERS



# Which Deployment Method Do I Choose?

## Which options do you have with SAS Viya?

Batch Processing



Process large amounts of data in batch mode workloads that may be very computationally intensive or are not a critical business task.

API Calls



Process data only when needed but have the flexibility to be called from anywhere. Depending on the workload you may need high throughput infrastructure.

Real-time Processing



Process large amount of data which are being generated all the time but not everything can be stored or is useful if not processed on-time.

Edge Computing



Process data which are being generated all the time by devices with low computing power which cannot be processed elsewhere due to cost or connection restrictions.

# Deployment with SAS Viya & Open Source

Engine	Technologies Deployed	How Model Deployment is Achieved	
Batch Processing	CAS	SAS	Even though only SAS models are deployed with scalability on CAS, you can develop and use these models using the SWAT packages in any language that SWAT supports.
API Calls	MAS	SAS Python	MAS serves models as API endpoints which can be used for single calls in high throughput context. The models served can be written in SAS or Python and they can be called from anywhere.
Batch Processing API Calls	Docker Kubernetes	Python R	Models deployed on Docker are in batch-like mode, but any R or Python code can be deployed and called from anywhere. If you deploy on Kubernetes, they can be also scalable (has limited functionality).
Real-time Processing Edge Computing	SAS ESP	SAS Python	Models deployed for real-time can be Python (through MAS) or SAS. Real-time engines are usually deployed next to the data source, sometimes being edge devices with low computing power. SAS ESP also allows you to manage its processes using Python with ESPPY.

# Deployment with SAS Viya & Open Source

## CAS

### When Should I Leverage CAS?

- You have large tables that can benefit usage of parallel and distributed environments
- Compute-intensive analytics which needs lots of processing power and time to run
- Tasks which run over time (scheduled)

Batch Processing

### How do I Access CAS?

- SAS (CASL and SAS Procedures\*)
- Python (SWAT)
- R (SWAT)
- LUA (SWAT)
- Java (CAS-Client)

### What Can I Deploy?

- CASL models/code (in any language)

\*When you run SAS code inside a SAS Viya environment, under the hood its procedures may be automatically translated to CAS, but it is not a rule since not all procedures have been rewritten.

# Deployment with SAS Viya & Open Source

## MAS

### When Should I Leverage MAS?

- You need a high-availability and high-throughput engine with API endpoints to be called from any other service
- You need a model to give a single answer that will be used for further actions elsewhere

API Calls

### How do I Access MAS?

- REST API
- [Python \(SASctl\)](#)
- SAS Model Manager (batch processing)
- SAS Event Stream Processing

### What Can I Deploy?

- Python Models
  - Requires specific code structure
  - It has to be wrapped into DS2 code
  - Can be generated from SASctl, Pzmm or manually
- SAS models
- [R/Py PMML Models](#)
  - PMML 4.2
  - Converted to SAS models

# Deployment with SAS Viya & Open Source

## Containers: Dockers and Kubernetes

### When Should I Leverage Containers?

- You need to deploy open source languages models
- Your models need high-availability and auto-scalable deployment (Kubernetes)
- Have a Docker image register service to help manage your containerized models

Batch  
Processing  
API Calls

### How do I Access Containers?

- REST API

When open source code is deployed to containers through SAS Model Manager, it will have a Flask webserver which will call your R or Python code.

Technically it will deploy any open source code that returns object like a “model output”.

As of the latest version (SAS Viya 3.5) containers are ideal for batch calls which must be called twice, one for scoring and other to retrieve results, both made through .csv files.

### What Can I Deploy?

- Python Models

- Requires specific code structure

- R Models

- Requires specific code structure

# Deployment with SAS Viya & Open Source

## SAS Jobs

### When Should I Leverage SAS Jobs?

- Existing SAS STPs need to be brought forward
- Lowest possible response time not required
- Required functionality not available in other options

### How do I Access SAS Jobs?

- REST API
  - (via sas-viya CLI)

### What Can I Deploy?

- Everything that SAS Compute supports (incl. calls to outside & 3<sup>rd</sup> party)



# Deployment with SAS Viya & Open Source

## SAS ESP

### When Should I Leverage SAS ESP?

Immense data points generated per timer unit that needs to be processed

Need an engine to be deployed anywhere, even edge devices (such as Raspberry pi)

Find important transactions and take actions inside a cascade of information that have value if processed when generated

Real-time Processing

Edge Computing

### How do I Access SAS ESP?

- ESPPy
  - Build projects
  - Retrieve and send data
- SAS ESP has a broad range of connectors and adaptors to tools such as databases, messaging systems, files etc.

### What Can I Deploy?

- Python Models
  - Through MAS
- SAS Models
  - Through MAS
- ESP Built-in Models
  - Real-time Training
- C/C++ Models
  - Compiled modules

# Publishing Destinations

Model source



SAS Cloud  
Analytics  
Service  
(CAS)



SAS Micro  
Analytics  
Service  
(MAS)



In-Stream



In-  
Database



In-  
Database



Git  
Repository



Azure  
Machine  
Learning



Container



Container



Container

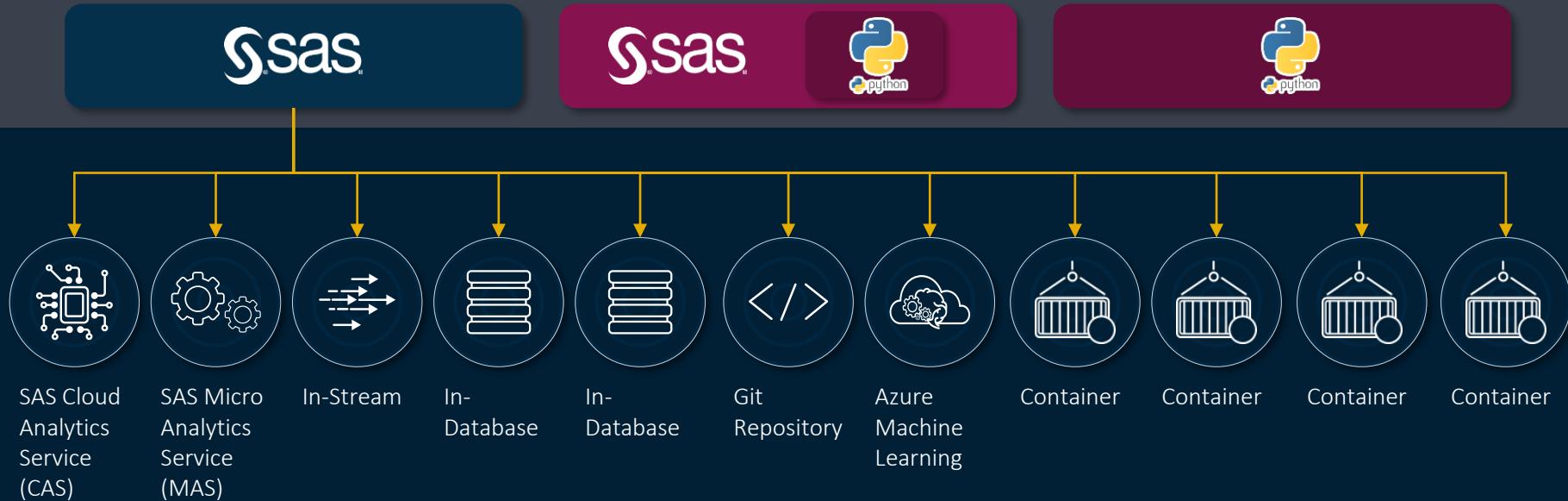


Container



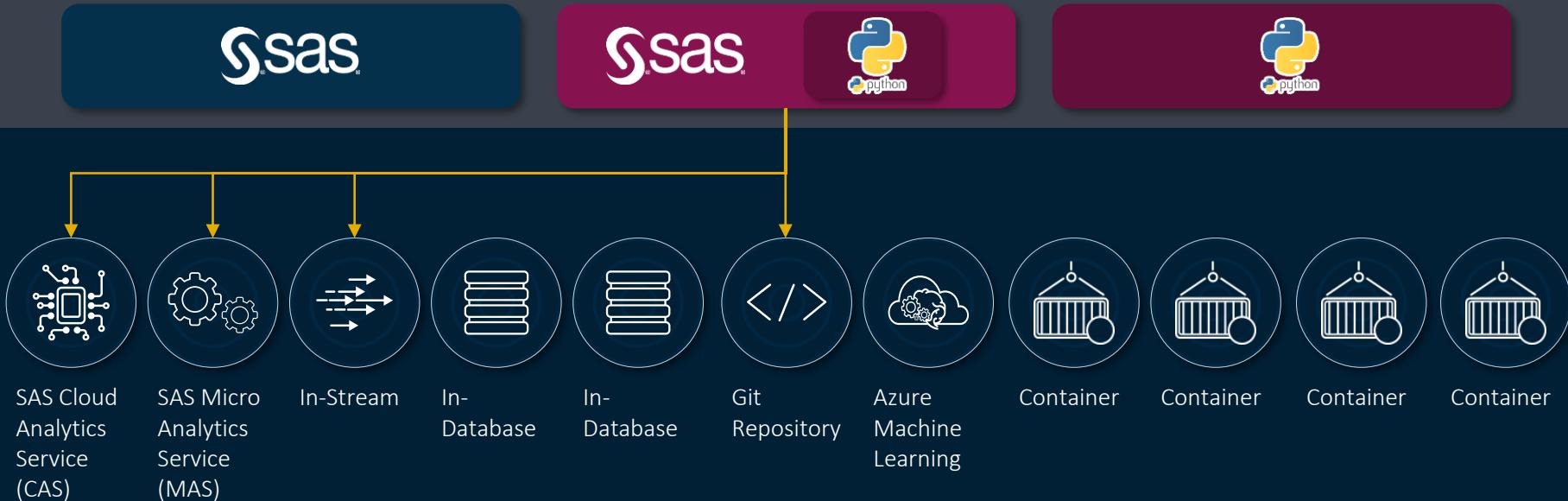
# Publishing Destinations

Model source



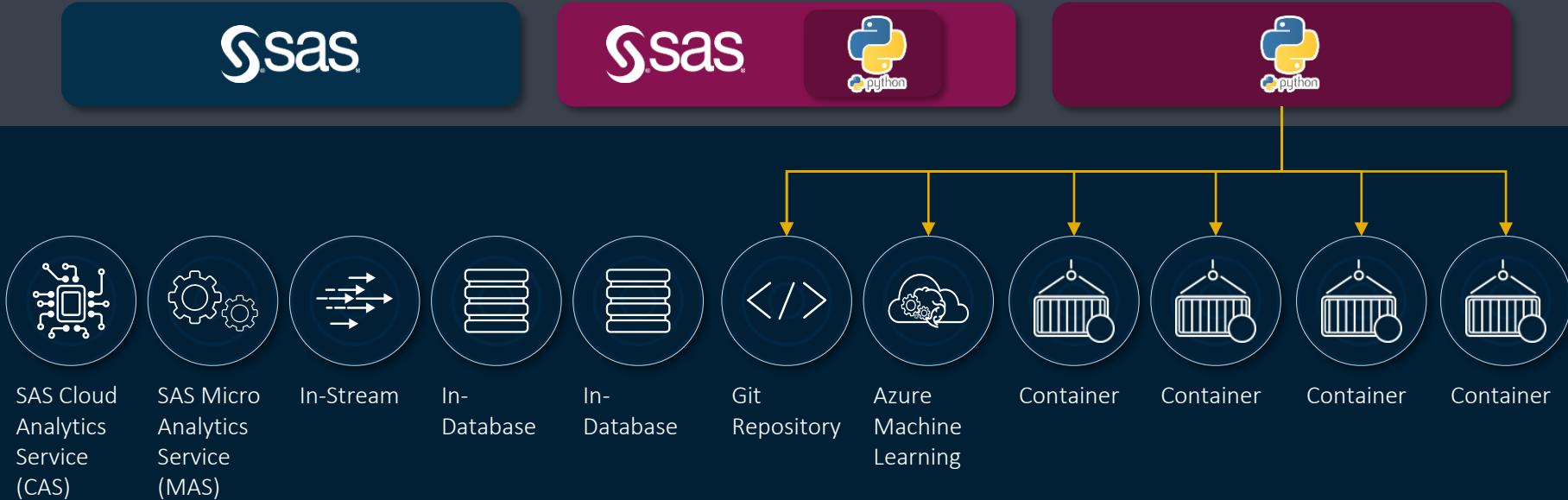
# Publishing Destinations

Model source



# Publishing Destinations

Model source





sas.com