

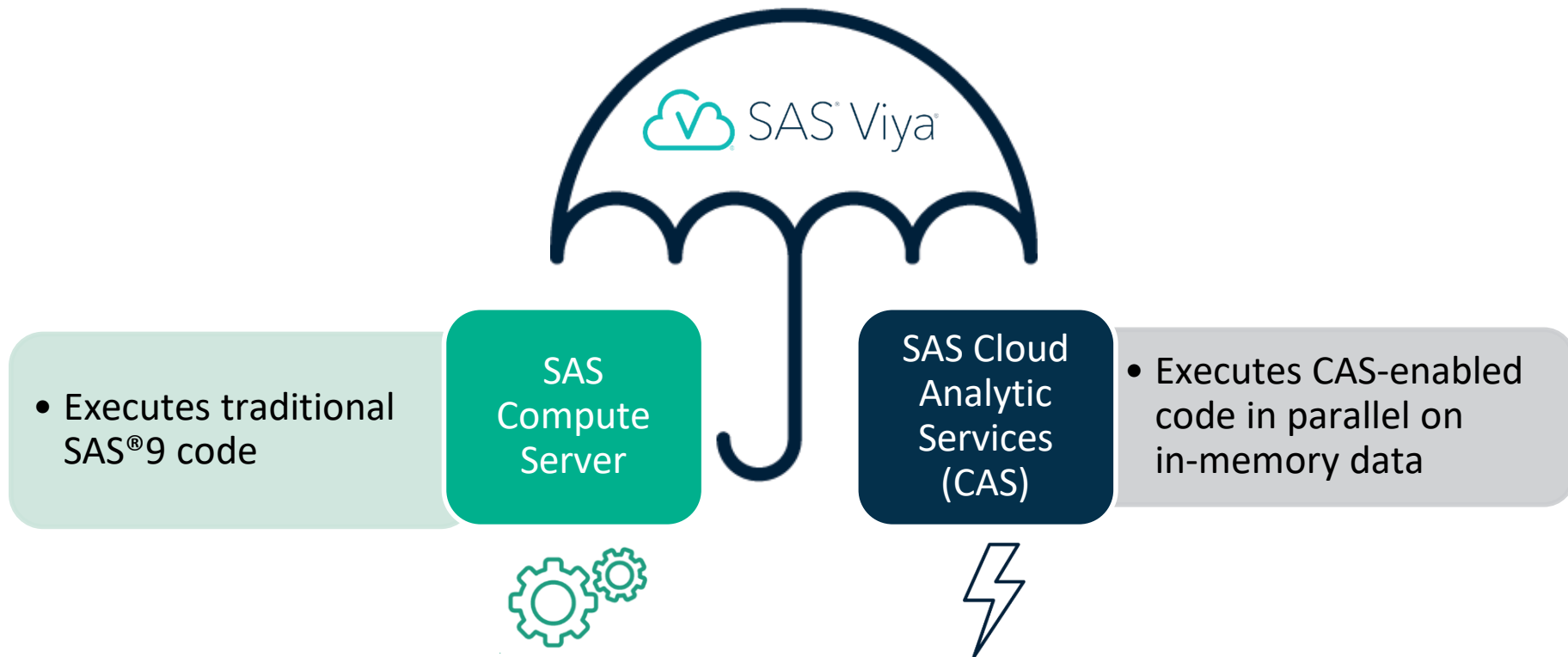
INITIATION À LA PROGRAMMATION SAS SUR UN SERVEUR CAS **CLUB DES UTILISATEURS SAS DE QUÉBEC**



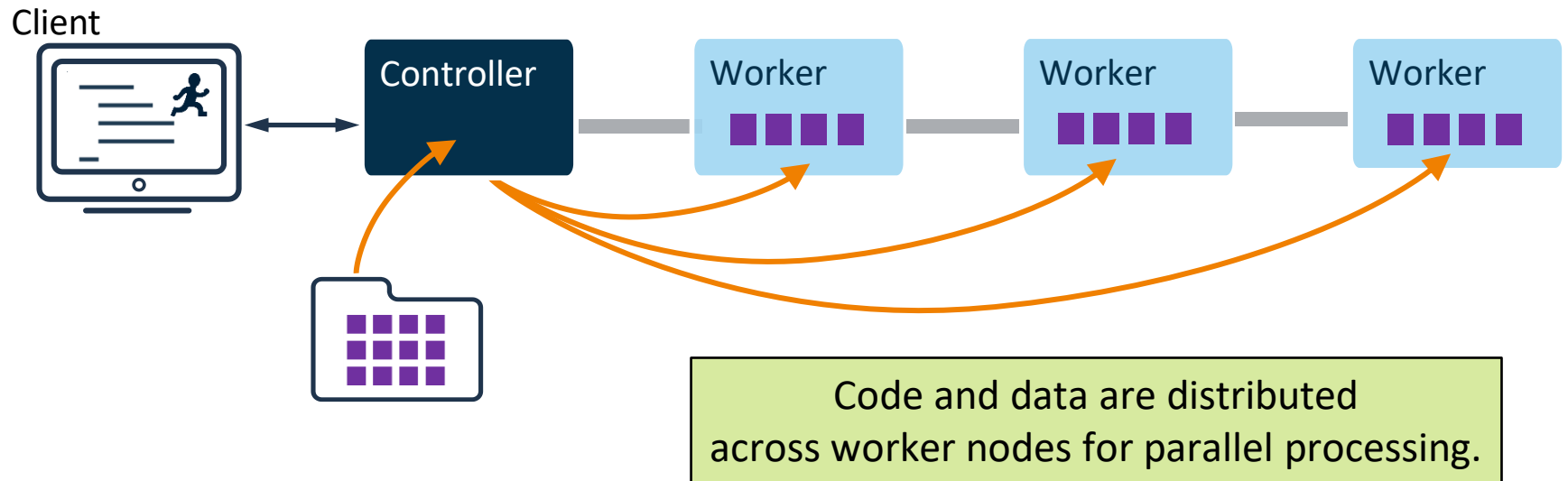
Sylvain Tremblay
SAS Canada

03 FÉVRIER 2022

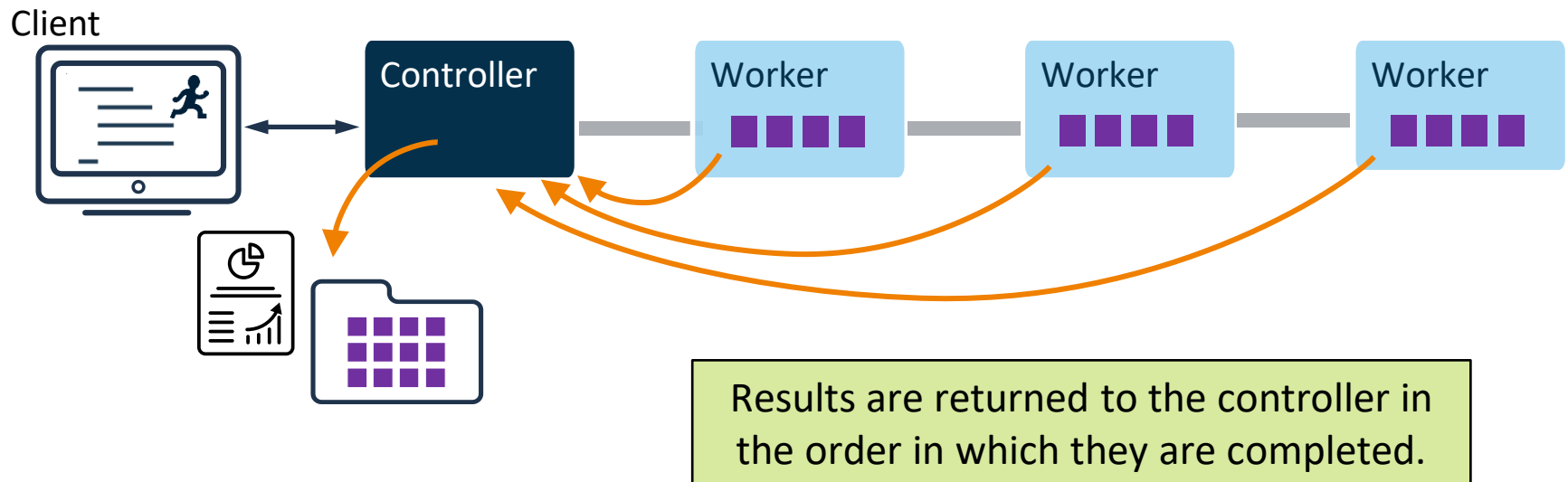
SAS Viya Servers



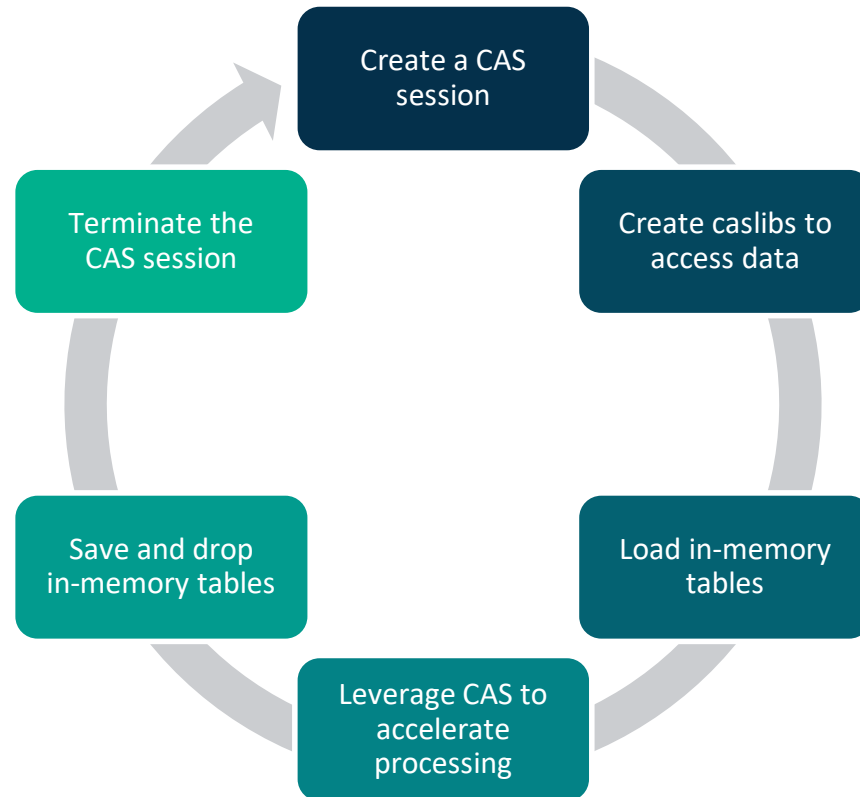
Parallel Processing in CAS



Parallel Processing in CAS



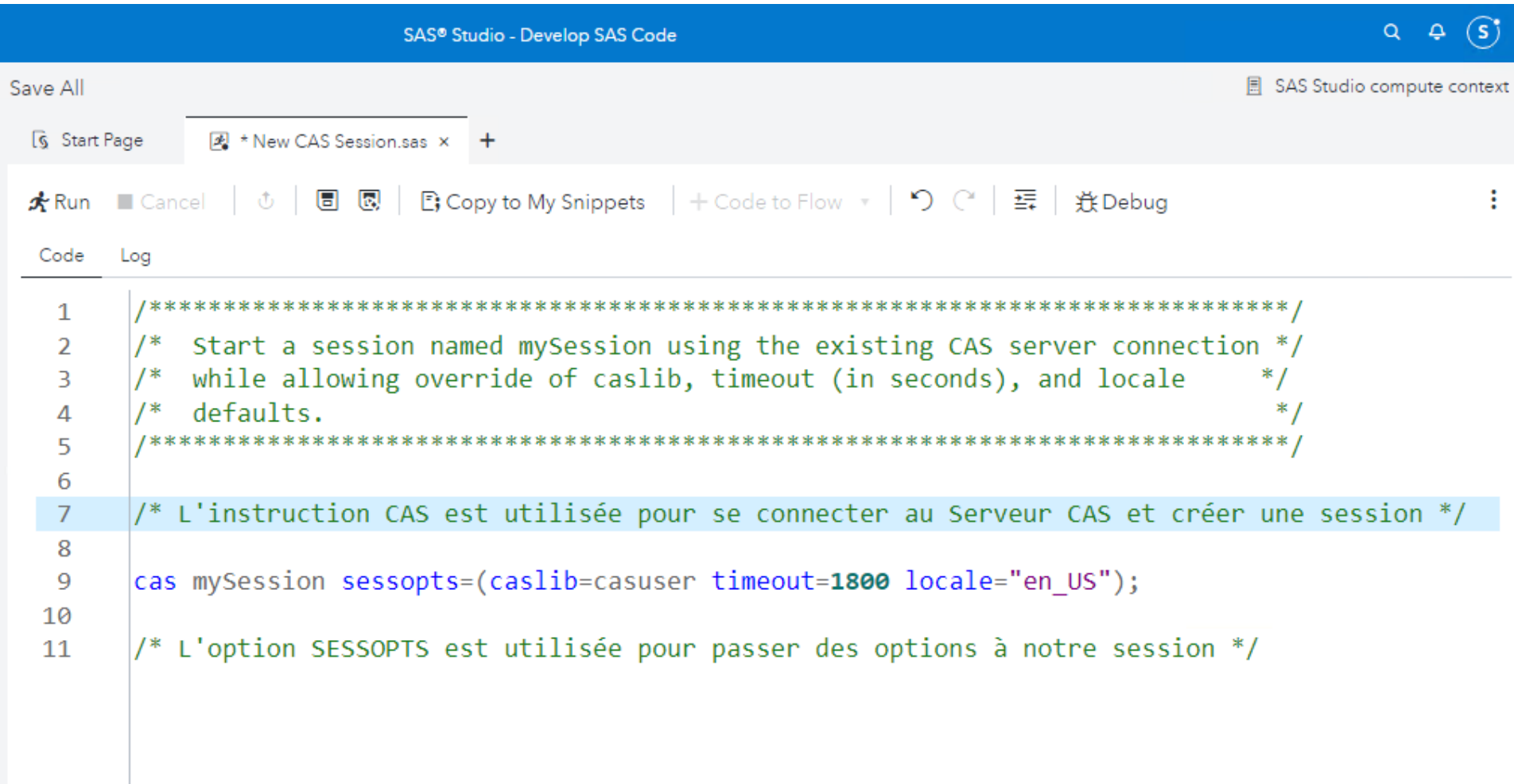
Processing Code in CAS



Agenda

- Création d'une session CAS
- Accès au données - CASLIB
- Chargement des données en mémoire
- Modification d'un programme Base SAS pour rouler sur CAS
- Prendre avantage de CAS: Procedures CAS et le langage CASL

Création d'une session CAS



The screenshot displays the SAS Studio interface. The title bar reads "SAS® Studio - Develop SAS Code". The top right corner contains search, refresh, and user icons. Below the title bar, a "Save All" button is on the left, and "SAS Studio compute context" is on the right. The editor area shows a file named "New CAS Session.sas". The toolbar includes "Run", "Cancel", "Copy to My Snippets", "Code to Flow", and "Debug". The code editor has tabs for "Code" and "Log". The code is as follows:

```
1  /*******/
2  /* Start a session named mySession using the existing CAS server connection */
3  /* while allowing override of caslib, timeout (in seconds), and locale */
4  /* defaults. */
5  /*******/
6
7  /* L'instruction CAS est utilisée pour se connecter au Serveur CAS et créer une session */
8
9  cas mySession sessopts=(caslib=casuser timeout=1800 locale="en_US");
10
11 /* L'option SESSOPTS est utilisée pour passer des options à notre session */
```

Création d'une session CAS

SAS® Studio - Develop SAS Code

Save All

SAS Studio compute context

Start Page * New CAS Session.sas x +

Run Cancel | Copy to My Snippets | Code to Flow | Debug Jan 19, 2022, 2:27:57 PM

Code Log

Errors (0) Warnings (0) Notes (6)

NOTE: The session MYSESSION connected successfully to Cloud Analytic Services

NOTE: The SAS option SESSREF was updated with the value MYSESSION.

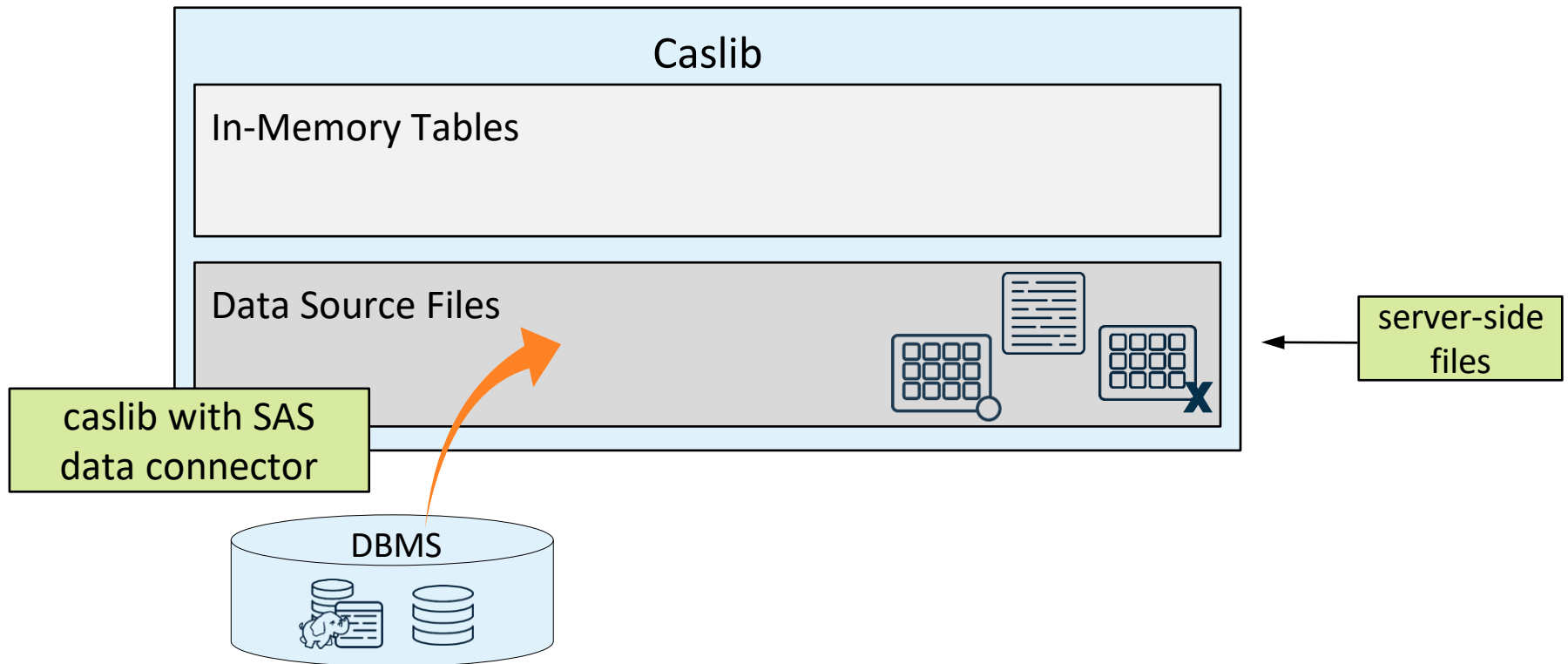
NOTE: The SAS macro _SESSREF_ was updated with the value MYSESSION.

```
79 /* Start a session named mySession using the existing CAS server connection */
80 /* while allowing override of caslib, timeout (in seconds), and locale */
81 /* defaults. */
82 /*****
83
84 /* L'instruction CAS est utilisée pour se connecter au Serveur CAS et créer une session */
85
86 cas mySession sessopts=(caslib=casuser timeout=1800 locale="en_US");
NOTE: The session MYSESSION connected successfully to Cloud Analytic Services
controller.sas-cas-server-default.edu.svc.cluster.local using port 5570. The UUID is 38b424f1-b599-014b-8985-ec2ab85fc5b6.
The user is student and the active caslib is CASUSER(student).
NOTE: The SAS option SESSREF was updated with the value MYSESSION.
NOTE: The SAS macro _SESSREF_ was updated with the value MYSESSION.
NOTE: The session is using 0 workers.
NOTE: 'CASUSER(student)' is now the active caslib.
NOTE: The CAS statement request to update one or more session options for session MYSESSION completed.
87
88 /* L'option SESSOPTS est utilisée pour passer des options à notre session */
89
90 %studio_hide_wrapper;
98
```


Agenda

- Création d'une session CAS
- Accès au données - CASLIB
- Chargement des données en mémoire
- Modification d'un programme Base SAS pour rouler sur CAS
- Prendre avantage de CAS: Procedures CAS et le langage CASL

Accès aux données - CASLIB



Accessing Data through Caslibs

SAS® Studio - Develop SAS Code

Save All

SAS Studio compute context

Start Page * New CAS Session.sas * New caslib for Path.sas x +

Run Cancel Copy to My Snippets Code to Flow Debug Jan 19, 2022, 2:48:15 PM

Code Log

```
1  /*****  
2  /* Create a CAS library (myCaslib) for the specified path ("/filePath/") */  
3  *****/  
4  
5  caslib myCaslib path="/home/student/Courses/PGVY01/data" libref=myCaslib;  
6  
7  /* Pour accéder aux tables qui vont résider en mémoire dans votre code */  
8  /* Vous devez d'abord assigner un LIBREF à votre CASLIB */  
9  /* Meilleure pratique: le nom de votre CASLIB et le nom de son LIBREF est le même */  
10
```

NOTE: 'MYCASLIB' is now the active caslib.
NOTE: Cloud Analytic Services added the caslib 'MYCASLIB'.
NOTE: CASLIB MYCASLIB for session MYSESSION will be mapped to SAS Library MYCASLIB.
NOTE: Action to ADD caslib MYCASLIB completed for session MYSESSION.

Libraries

- Libraries
 - MAPS
 - MAPSGFK
 - MAPSSAS
 - MYCASLIB

```
/* Pour voir les fichiers disponibles dans votre CASLIB */
```

```
Proc casutil incaslib=myCaslib;  
  list files;  
quit;
```

Code Log Results

📁 The Casutil Procedure

📄 CaslibInfo

📄 File Information for root of casli...

The CASUTIL Procedure

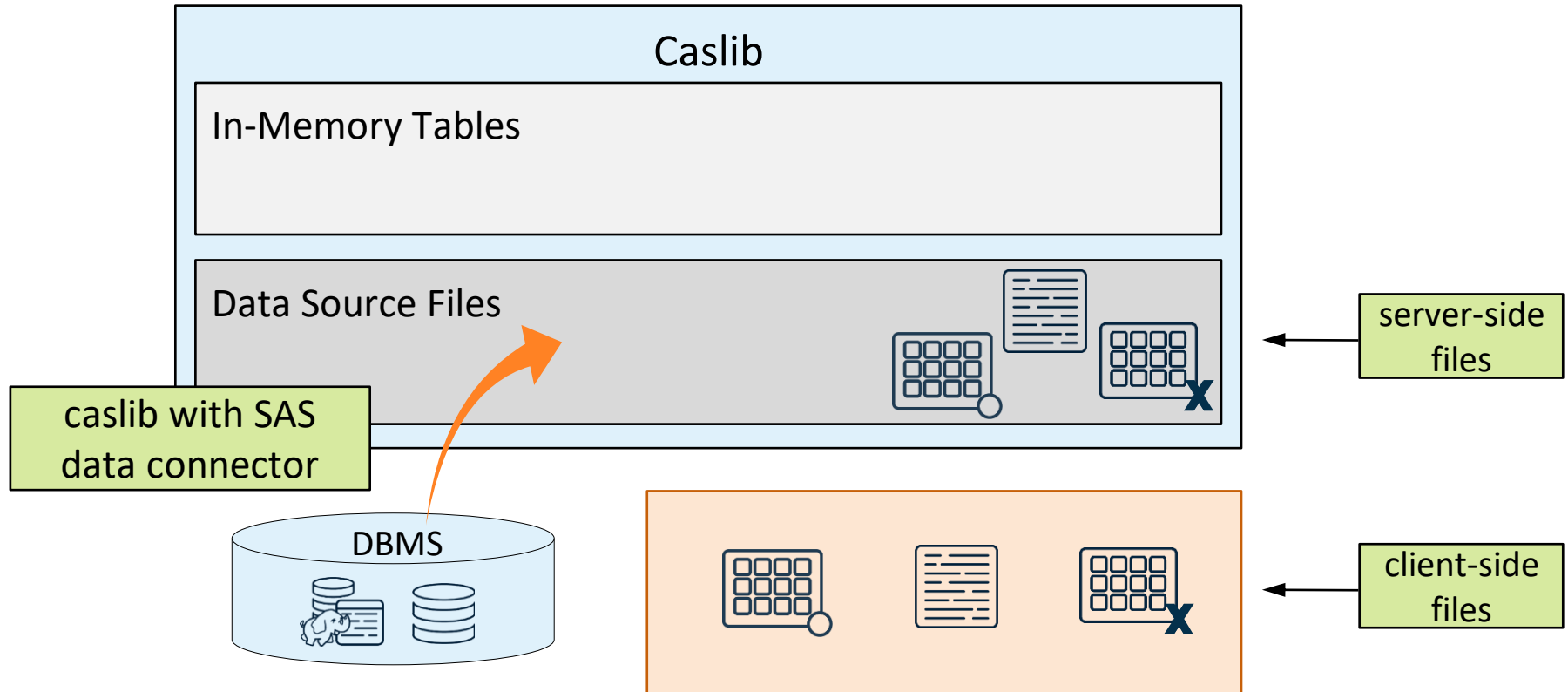
CAS File Information

Name	Permission	Owner	Group	Encryption Method	File Size	Last Modifie (UTC)
country_lookup.sas7bdat	-rwxrwxrwx				72.0KB	09JUN2020:
customers.xlsx	-rwxrwxrwx				4.1MB	15JUN2020:
employees.sas7bdat	-rwxrwxrwx				256.0KB	12JUN2020:
orders.csv	-rwxrwxrwx				245.2MB	11JUN2020:
orders.sas7bdat	-rwxrwxrwx				1.6GB	12JUN2020:
orders.xlsx	-rwxrwxrwx				130.1MB	11JUN2020:
orders_new.sas7bdat	-rwxrwxrwx				17.5MB	13JUN2020:
products.sas7bdat	-rwxrwxrwx				1.8MB	18JUN2020:
qtr_sales.sas7bdat	-rwxrwxrwx				2.7MB	12JUN2020:
sales.xlsx	-rwxrwxrwx				21.4KB	13JUN2020:
create_bigOrders.sas	-rwxr-xr-x				0.5KB	08DEC2020
bigorders.sas7bdat	-rw-r--r--				16.1GB	08DEC2020

Agenda

- Création d'une session CAS
- Accès au données - CASLIB
- Chargement des données en mémoire
- Modification d'un programme Base SAS pour rouler sur CAS
- Prendre avantage de CAS: Procedures CAS et le langage CASL

Chargement des données en mémoire



Chargement des “Server-Side Files”

```
proc casutil;  
  load casdata="orders.sas7bdat"  
    incaslib="myCaslib"  
    outcaslib="myCaslib"  
    casout="commandes";  
quit;
```

NOTE: Cloud Analytic Services made the file orders.sas7bdat available as table COMMANDES in caslib myCaslib.

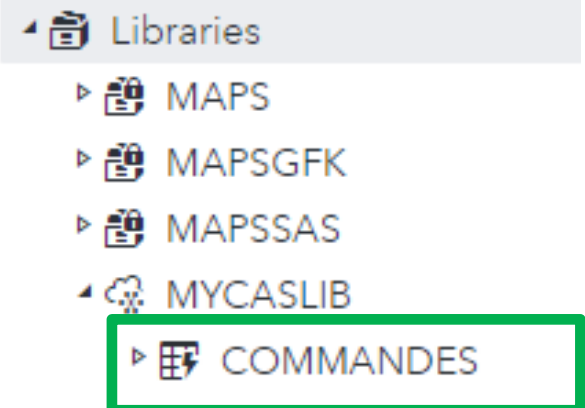
NOTE: The Cloud Analytic Services server processed the request in 10.364512 seconds.

83 quit;

NOTE: PROCEDURE CASUTIL used (Total process time):

real time 10.36 seconds

cpu time 0.01 seconds



Libraries

- MAPS
- MAPSGFK
- MAPSSAS
- MYCASLIB
 - COMMANDES

Agenda

- Création d'une session CAS
- Accès au données - CASLIB
- Chargement des données en mémoire
- Modification d'un programme Base SAS pour rouler sur CAS
- Prendre avantage de CAS: Procédures CAS et le langage CASL

Modification d'un programme Base SAS

Pour rouler sur CAS

Une étape DATA peut être exécuté sur le serveur CAS si:

- Les bibliothèques référencées sont des CASLIB
- Les éléments de langage sont supportés par CAS

Certaines Procédures de Base SAS sont CAS enabled

Etape Data pouvant être exécutée sur CAS

```
/* Etape Data executee sur le Serveur CAS */
```

```
.....  
Data myCaslib.Berlin_orders;  
    set myCaslib.commandes;  
    where Continent="Europe" and City="Berlin";  
run;
```

```
78  Data myCaslib.Berlin_orders;  
79      set myCaslib.commandes;  
80      where Continent="Europe" and City="Berlin";  
81  run;
```

NOTE: Running DATA step in Cloud Analytic Services.

NOTE: The DATA step will run in multiple threads.

NOTE: There were 7652 observations read from the table COMMANDES in caslib MYCASLIB.

NOTE: The table Berlin_orders in caslib MYCASLIB has 7652 observations and 24 variables.

NOTE: DATA statement used (Total process time):

real time 0.08 seconds

cpu time 0.01 seconds

Etape Data ne pouvant pas être exécutée sur CAS

```
/* Etape Data non executee sur le Serveur CAS */  
Data work.Paris_orders(when=(RetailPrice > 100));  
    set myCaslib.commandes;  
    where Continent="Europe" and City="Paris";  
    format Cost Words20.;  
run;
```

```
78 Data work.Paris_orders(when=(RetailPrice > 100));  
79     set myCaslib.commandes;  
80     where Continent="Europe" and City="Paris";  
81     format Cost Words20.;  
82 run;
```

Compute
Server

```
NOTE: There were 34597 observations read from the data set MYCASLIB.COMMANDES.  
      WHERE (Continent='Europe') and (City='Paris');  
NOTE: The data set WORK.PARIS_ORDERS has 14071 observations and 24 variables.  
NOTE: DATA statement used (Total process time):  
      real time          1.04 seconds  
      cpu time           0.30 seconds
```

Certaines Procedures Base SAS sont CAS enabled

```
/* Procedure SAS CAS Enabled */  
Proc means data=myCaslib.Berlin_orders;  
    var Quantity Cost;  
run;
```

```
78 Proc means data=myCaslib.Berlin_orders;  
79     var Quantity Cost;  
80 run;
```

NOTE: The CAS aggregation.aggregate action will be used to perform the initial summarization.

NOTE: The PROCEDURE MEANS printed page 1.

NOTE: PROCEDURE MEANS used (Total process time):

real time	0.12 seconds
cpu time	0.02 seconds

La table lue est in-memory, la syntaxe est CAS compliant. SAS va convertir le code en Actions CAS

Certaines Procedures Base SAS sont CAS enabled

```
/* Procedure SAS CAS Enabled non executee sur le Serveur CAS */  
Proc means data=myCaslib.Berlin_orders mean median min max;  
    var Quantity Cost;  
run;
```

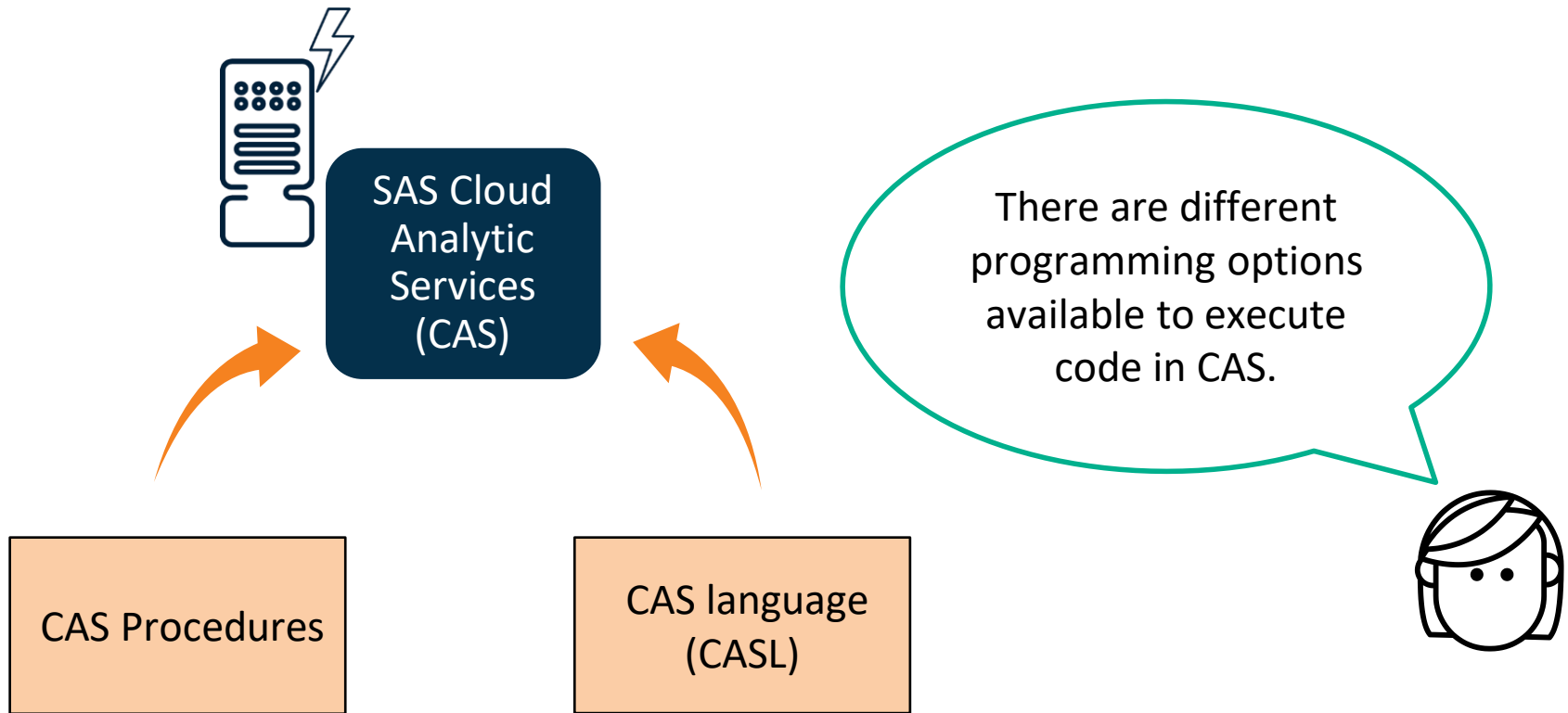
```
78 Proc means data=myCaslib.Berlin_orders mean median min max;  
79     var Quantity Cost;  
80 run;  
NOTE: There were 3723 observations read from the data set MYCASLIB.BERLIN_ORDERS.  
NOTE: The PROCEDURE MEANS printed page 2.  
NOTE: PROCEDURE MEANS used (Total process time):  
      real time           0.04 seconds  
      cpu time            0.03 seconds
```

La table lue est in-memory, la syntaxe est non CAS compliant. L'exécution a lieu sur le Compute Server

Agenda

- Création d'une session CAS
- Accès au données - CASLIB
- Chargement des données en mémoire
- Modification d'un programme Base SAS pour rouler sur CAS
- Prendre avantage de CAS: Procédures CAS et le langage CASL

Prendre avantage du serveur CAS



Prendre avantage du serveur CAS

Procedures CAS

Machine Learning

- FACTMAC (Factorization Machine Model)
- FOREST (Forest Model)
- GRADBOOST (Gradient Boosting Model)
- NNET (Neural Network)
- SVMACHINE (Support Vector Machine)
- SVDD (Support Vector Data Description)
- BNET (Bayesian Network)
- BOOLRULE (Boolean Rules)
- FASTKNN (k-nearest neighbor)
- GVARCLUS (Variable Clustering and Graphical Modeling)
- MBANALYSIS (Association Rule Mining)
- RPCA (Robust Principal Component Analysis)

Prendre avantage du serveur CAS

Procedures CAS

```
/* Prendre avantage du Serveur CAS - Procedures CAS */  
Proc MDSummary data=myCaslib.commandes;  
  var Quantity Cost;  
  groupby Continent;  
  output out=myCaslib.orders_stat;  
run;
```

Continent_f	_Column_	_Min_	_Max_
Africa	Quantity	1	6
Africa	Cost	0.5	1552.6
Asia	Quantity	1	6
Asia	Cost	1.2	579.9
Europe	Quantity	1	10
Europe	Cost	0.4	1583.6

```
78 Proc MDSummary data=myCaslib.commandes;  
79   var Quantity Cost;  
80   groupby Continent;  
81   output out=myCaslib.orders_stat;  
82 run;
```

NOTE: The Cloud Analytic Services server processed the request in 0.052905 seconds.

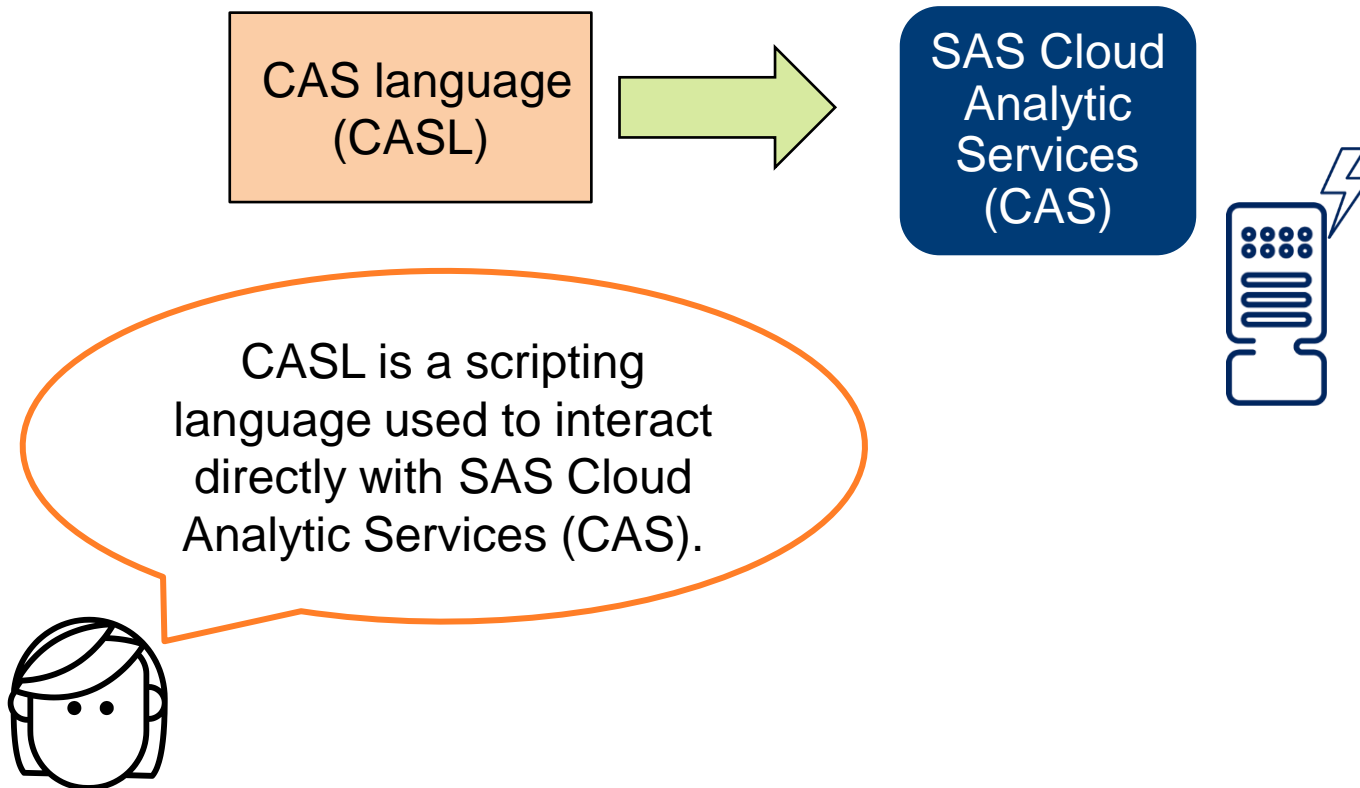
NOTE: The data set MYCASLIB.ORDERS_STAT has 10 observations and 19 variables.

NOTE: PROCEDURE MDSUMMARY used (Total process time):

real time	0.07 seconds
cpu time	0.00 seconds

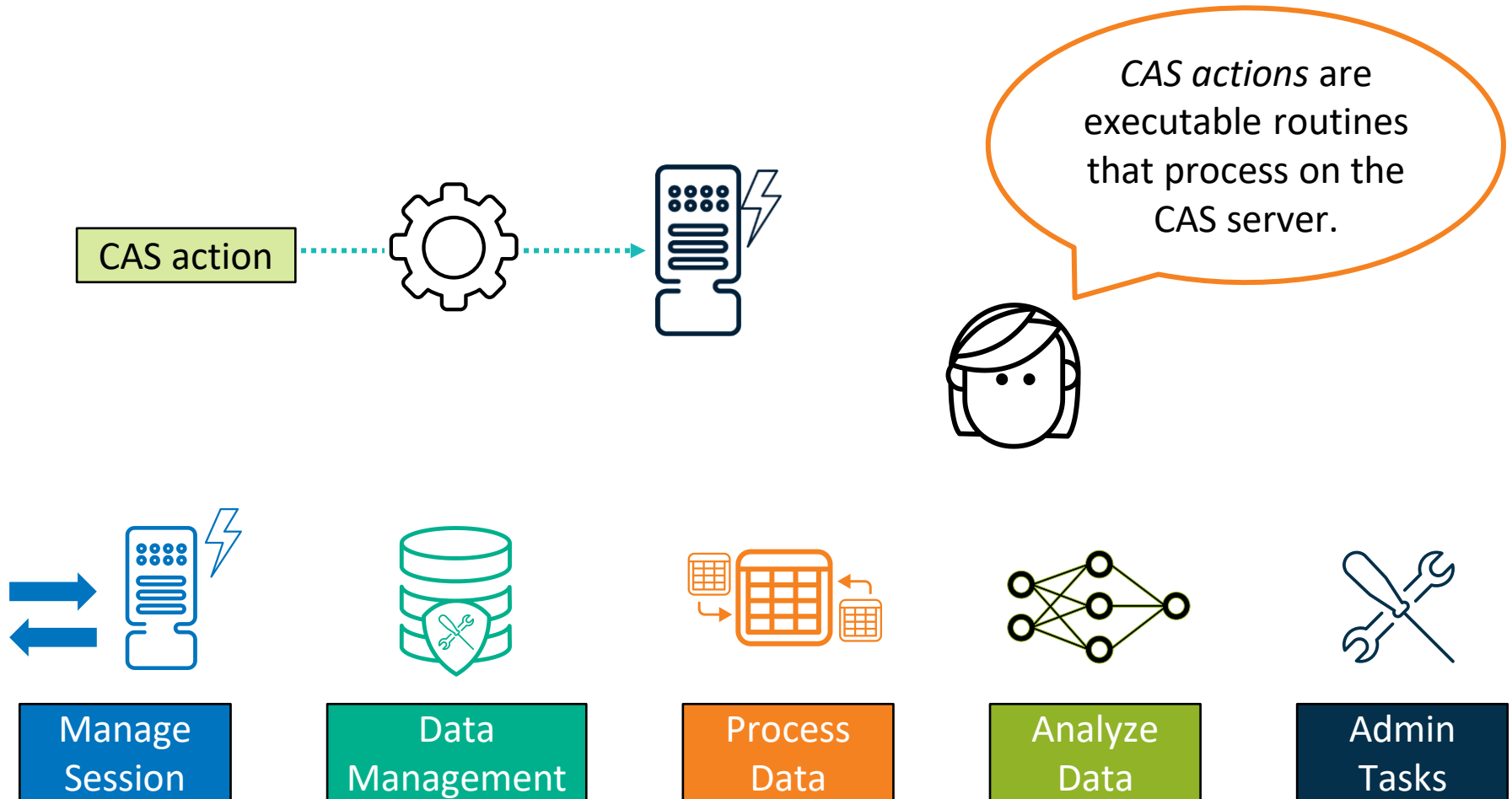
Prendre avantage du serveur CAS

Language CASL



Prendre avantage du serveur CAS

Language CASL



Prendre avantage du serveur CAS

Language CASL

Simple Analytics Action Set

```
proc cas;  
  simple.distinct /  
    table={name="sales", caslib="casuser"},  
    input={"Job Title", "Country"};  
quit;
```

Distinct Counts for SALES			
Column	Number of Distinct Values	Number of Missing Values	Truncated
Job Title	5	0	No
Country	1	0	No

Prendre avantage du serveur CAS

Language CASL

Why should I
learn CASL?



Speak the native language of the CAS server

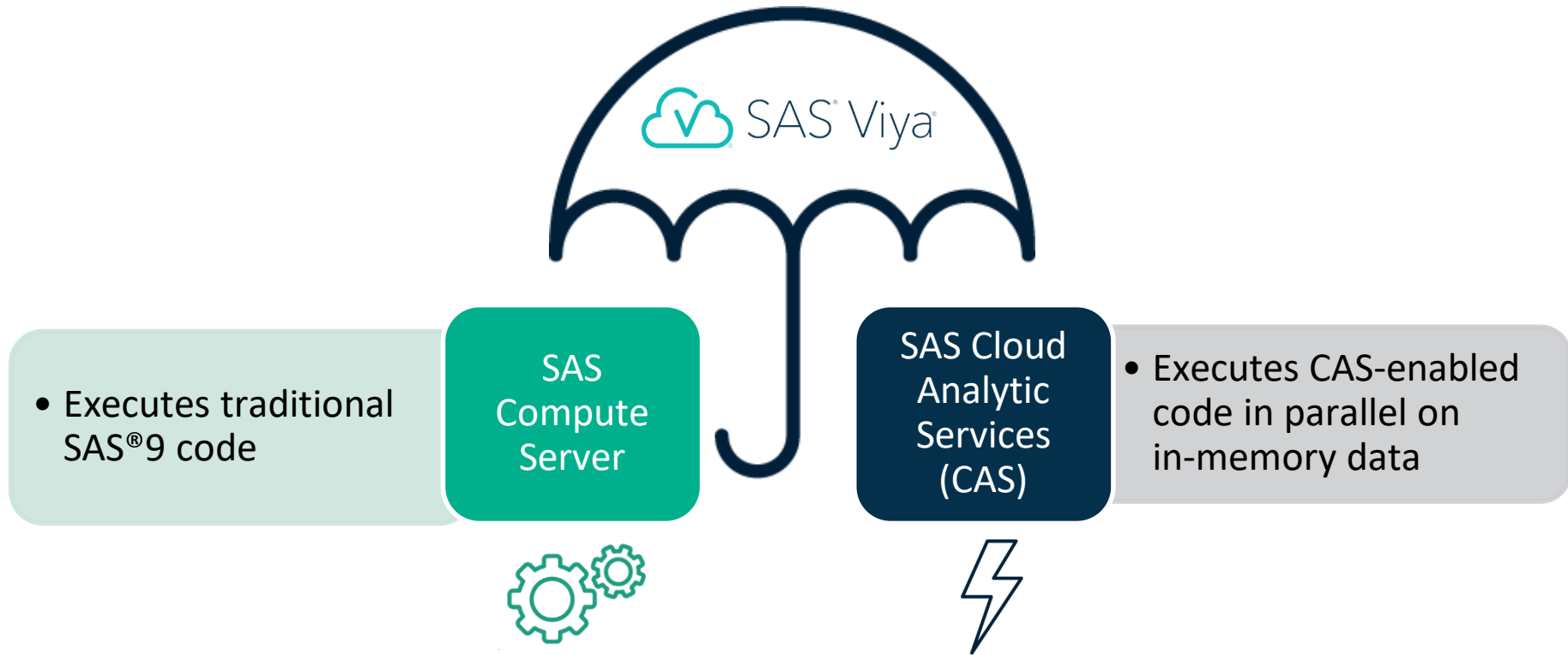


Submit only the actions that you need



Take advantage of actions and parameters not available in CAS-enabled steps

En Conclusion



Références – Liens URL

- [A Beginner's Guide to Programming in the SAS Cloud](#)
- [Best Practices for Converting SAS Code to Leverage CAS](#)
- [Running the DATA Step in CAS](#)
- [Base SAS procedures can run CAS actions](#)

Merci!



**THE
POWER
TO KNOW®**

Sylvain.Tremblay@sas.com