Paper SAS1812-2015

Hey! SAS® Federation Server Is Virtualizing 'Big Data'!

Ivor G. Moan, SAS Institute Inc.

ABSTRACT

In this session, we discuss the advantages of SAS® Federation Server and how it makes it easier for business users to access secure data for reports and use analytics to drive accurate decisions. This frees up IT staff to focus on other tasks by giving them a simple method of sharing data using a centralized, governed, security layer. SAS Federation Server is a data server that provides scalable, threaded, multi-user, and standards-based data access technology in order to process and seamlessly integrate data from multiple data repositories. The server acts as a hub that provides clients with data by accessing, managing, and sharing data from multiple relational and non-relational data sources as well as from SAS® data. Users can view data in big data sources like Hadoop, SAP HANA, Netezza, or Teradata, and blend them with existing database systems like Oracle or DB2. Security and governance features, such as data masking, ensure that the right users have access to the data and reduce the risk of exposure. Finally, data services are exposed via a REST API for simpler access to data from third-party applications.

INTRODUCTION

What is Big Data and how can SAS Federation Server help?

Gartner's big data definition is not much longer than a tweet: "Big data" is high-volume, -velocity and -variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making.

Let's be honest: most people still do not have their heads around 'Big Data'! Hasn't data always been big? Big is of course all relative – one person's big is another person's small! Essentially it's all a matter of perspective: a star in the sky is a useful guiding light on the ocean at night, but how many people really conceive the reality that this point of light might have taken millions of years to reach our eye and that the distance between us and the star is really BIG!

In terms of data, volumes are projected to grow past the point that normal people can conceive the size. In fact many argue that this, in many cases, has already occurred. What then can we do when it becomes so overwhelming? The fundamental is that we have to gain control – we have to abstract the enormous – we have to exploit the systems that have been explicitly designed to scale to meet the challenge. Enter SAS Federation Server.

With SAS Federation Server, Big Data Virtualization is a key theme. Whether the Big Data takes the form of an SAP HANA appliance, or a Netezza appliance, or a Hadoop cluster, SAS Federation Server has it covered. In fact when you see the data for the first time it looks strangely familiar – just like any other data… and that is exactly how it should be! We abstract the huge distances that stars are away from us to simple points of guiding light in the sky – in the same way, SAS Federation server allows us to abstract huge volumes of data into collections, libraries, tables even though the underlying reality is quite different. But isn't that how it should be?

WHY DATA FEDERATION/VIRTUALIZATION?

Data Federation is a faster way to exploit existing business information with minimal data movement. It seamlessly enables secure access to heterogeneous databases and applications.

Secure access to data is of utmost priority. Lack of controlled & monitored access to real-time data from disparate systems leads to data breach incidents. The cost per incident in Germany & US alone highlights the business issue:

Globally, the cost of a data breach averaged $136 per compromised record, Total Cost of per Incident in Germany was US $4.8m and $5.4m in US. *Ponemon Institute and Symantec, June 2013*

Without data virtualization, you risk knowing less about your customers…fewer real-time business insights, lose your competitive advantage, and spend more to address data challenges…*Forester, 2013 – Forester Information Fabric*

Security & Audit requirements are here to stay and going to get ever so more complicated. Un-monitored data access is a luxury that businesses can no longer afford. Organizations will be thinking twice if not three times on the idea of giving away 2% of Global Sales in penalties.

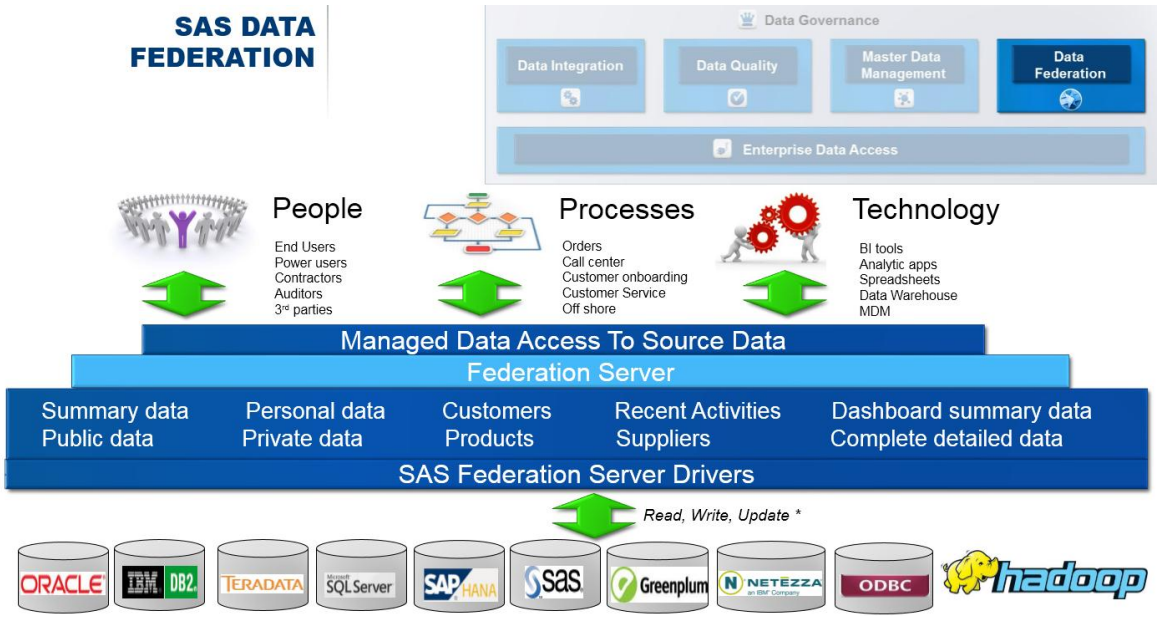


Figure 1. SAS Data Federation architecture

SAS Federation Server empowers business users to access and manage their own data. It provides a virtual layer or view, giving users the appropriate level of control without physically moving data.

By accessing data virtually, there is no need to move data, replicate it or retrieve it from tables to perform analysis.

Key Benefits

- Provides a data virtualization layer that enables consistent access to multiple underlying data sources with a single connection
- Simplifies data maintenance and administration
- Enables different security levels for different users
- Retrieves result data quickly from underlying data sources
- Real time access to data via web services allows for data to be readily available for decision making purposes delivering value more quickly
- Reduces Data Integration costs
- Reduces dependency on IT to get access to data needed for analysis

HOW IS SAS FEDERATION SERVER VIRTUALIZING BIG DATA?

SAS Federation Server Data Sources include the following in which SAP HANA, IBM Netezza, and Hadoop are recognized Big Data systems:

Base SAS		Teradata	
DB2 CS		Greenplum	
Oracle		Netezza	
SAP		SASHDAT	
SAP HANA		PostgreSQL	
SQL Server		Hadoop HIVE	
ODBC		Impala	

Figure 2. SAS Federation Server Data Sources

Big Data systems are characterized not only by the volumes of data that can be present, but also by the diversity and the processing power that can be brought to bear on that data. It's not an accident that all three of the mentioned systems (SAP HANA, IBM Netezza and Hadoop) are implemented on appliances that are highly scalable in both the processing and storage dimensions.

We can define 'Data Services' in a quite straight forward way using the SAS Federation Server Manager Web interface. We can define Data Source Names, which we reference in our client applications, to each 'Data Service' that determines how we access each service (for example, the SQL dialect):

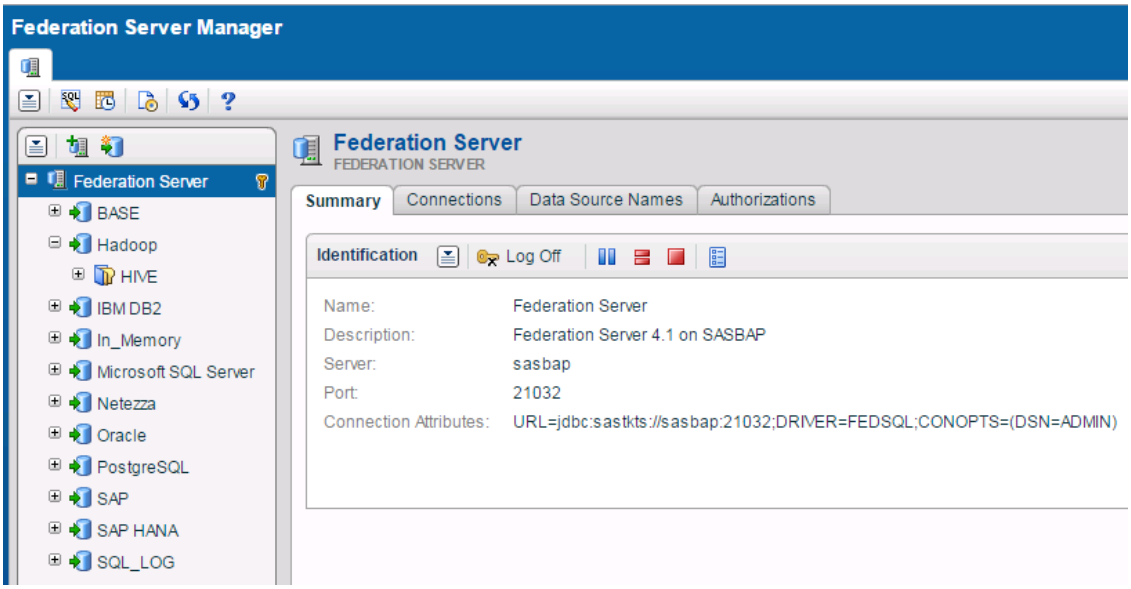


Figure 3. The SAS Federation Server Manager

It has to be pointed-out, at this point, that the SAS Federation Server Manager is very much the tool of an administrator and most likely a member of an IT department. The administrator defines the virtual layer that is Federation Server; makes definitions that control the access; makes definitions that determine what is being accessed and is involved in managing/monitoring these items.

For example one group of users can access a Hadoop environment with the following SAS libref:

```
LIBNAME FedHDP FEDSVR DSN=Hadoop PRESERVE_TAB_NAMES=YES catalog=HIVE
SERVER="sasbap" PORT=21032 USER="sasbap\sasdemo" PASSWORD="<password>";
```

…and see the following tables:

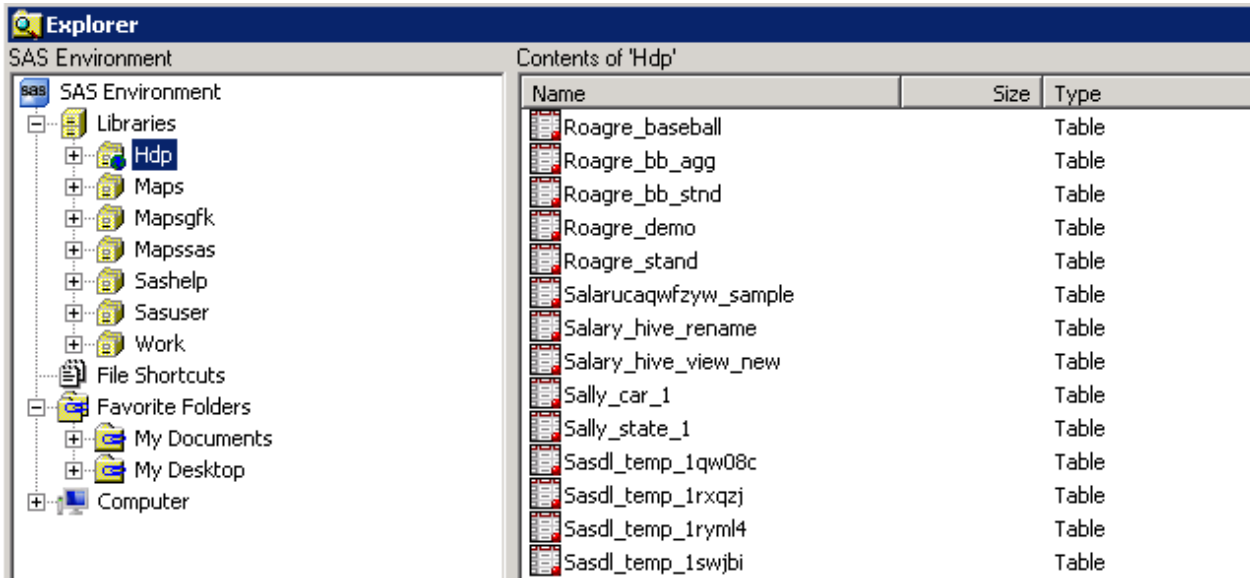


Figure 4. The SAS Explorer showing a Federation Server library surfacing Hadoop data

…and another group of users can access a Hadoop environment with the same libref and see the following:

```
LIBNAME FedHDP FEDSVR DSN=Hadoop PRESERVE_TAB_NAMES=YES catalog=HIVE
SERVER="sasbap" PORT=21032 USER="sasbap\sastest" PASSWORD="<password>";
```

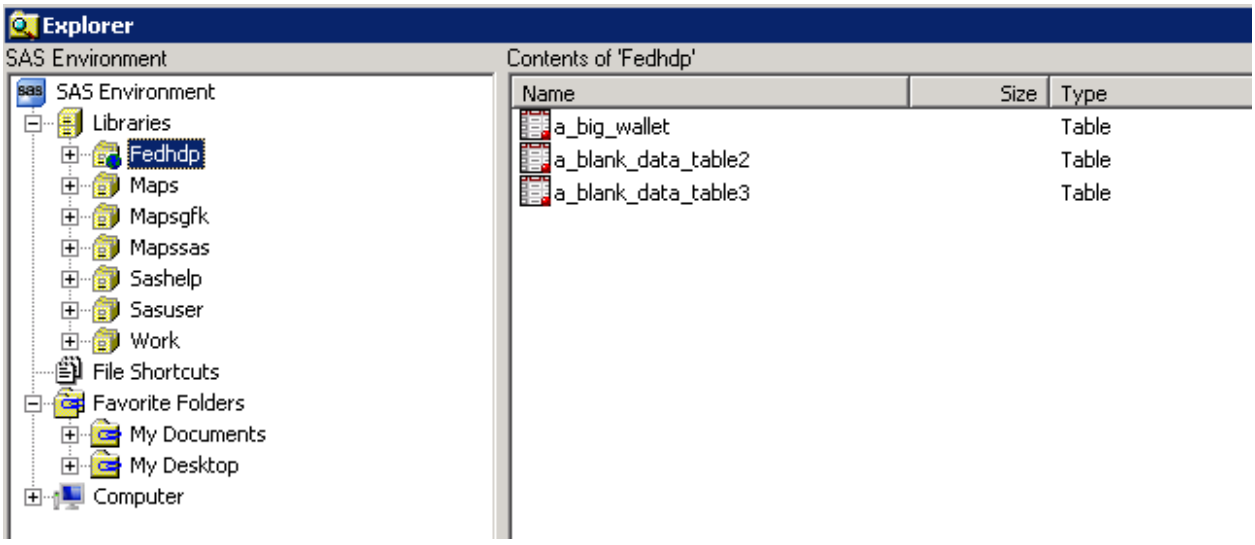


Figure 5. The SAS Explorer showing a Federation Server library surfacing Hadoop data

This allows SAS Federation Server to act as a security layer and gateway to non-secured Hadoop systems – with the logging capability and log reports we can see who is doing what and when they are doing it.

I already made the remark that “SAS Federation server allows us to abstract huge volumes of data into collections, libraries, tables even though the underlying reality is quite different. But isn’t that how it should be?” – Doesn’t this seem easy?

A key feature of SAS Federation Server is the ability for us to create FedSQL views. FedSQL is an SQL processor that is ANSI 99-core compliant. FedSQL can be exploited from SAS using PROC FedSQL or from the SAS Federation Server where it provides a common SQL syntax across all the data sources. The FedSQL processor feature set includes many new data types such as integer and varchar, a multi-threaded execution pipeline, extensive code generation and an optimizer that supports Federated Implicit Pass-Through and heterogeneous joins.

Closely related to the concepts of Implicit Pass-Through and heterogeneous joins is the concept of federated query decomposition. The objective of this concept is to decompose a query in an optimal manner for the purposes of federated query processing.

Here we see a simple FedSQL view using implicit pass-through, where the processing is done on Hadoop and only the result data is surfaced to the SAS Federation Server – a key thing to point out with Big Data is that you do not want to move it if you don’t need to:

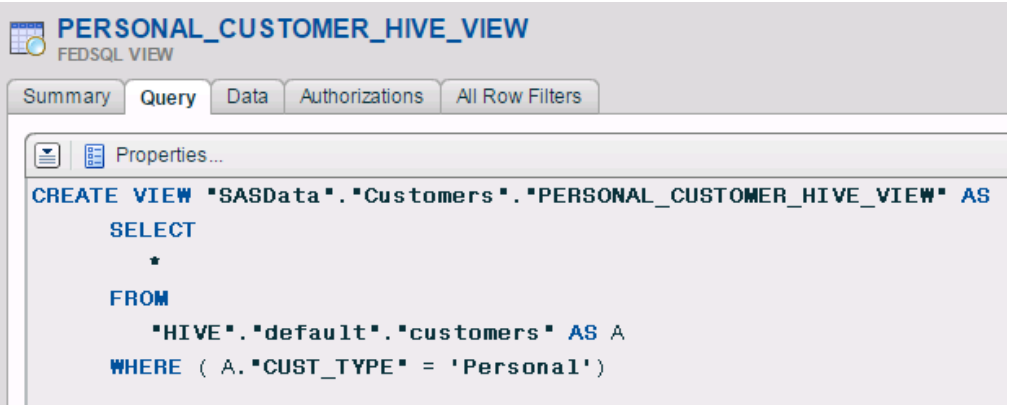


Figure 6. The definition of a FEDSQL view

Assuming that this data was only changing on a daily basis and that many people were accessing the view, we could cache this in SAS Federation Server and schedule the view for a daily refresh:

Figure 7. The FEDSQL view summary

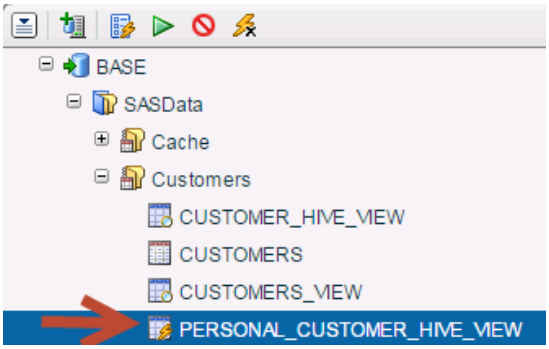


Figure 8. A cached FEDSQL view

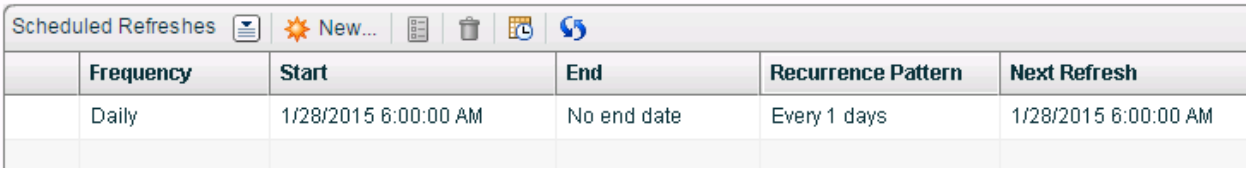


Figure 9. A scheduled refresh for a cached FEDSQL view

After the FedSQL view is cached, when we subsequently access it the data, unnecessary processing and data movement is prevented and significant improvements on access times should be achieved.

Things move fast when we hit the cache:

```
1    LIBNAME Fedsas FEDSVR DSN=Company SERVER="sasbap" catalog=SASData Schema=Customers PORT=21032
1  ! USER="sasbap\sasdemo" PASSWORD=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX;

NOTE: Libref FEDSAS was successfully assigned as follows:
      Engine:        FEDSVR
      Physical Name: Company

2    data customer;
3       set Fedsas.Personal_Customer_HIVE_View;
4    run;

NOTE: There were 4932 observations read from the data set FEDSAS.PERSONAL_CUSTOMER_HIVE_VIEW.
NOTE: The data set WORK.CUSTOMER has 4932 observations and 35 variables.
NOTE: DATA statement used (Total process time):
      real time           1.53 seconds
      cpu time            0.93 seconds
```

Figure 10. A SAS log showing data being read from our cached FEDSQL view

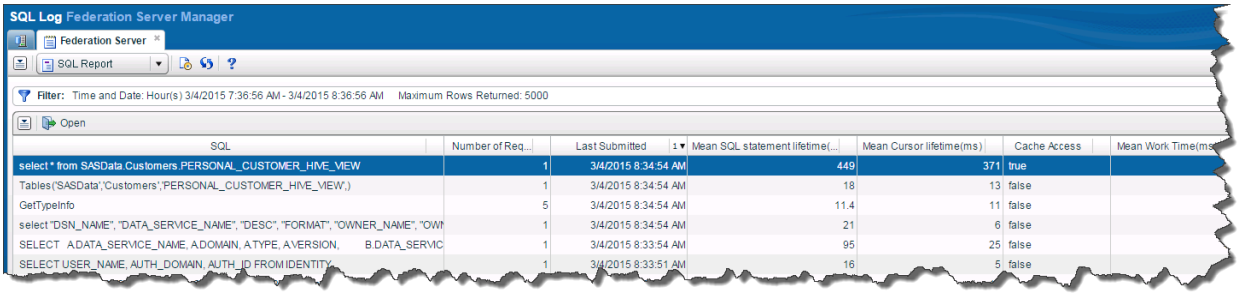


Figure 11. The Federation Server log showing the corresponding SQL report

When we think about customer data we normally also think about security – who is allowed to see Personally Identifiable Information (PII)? We need to comply with privacy laws and mask this data at the source. With SAS Federation Server we can build this feature into the data views and ensure encryption that can only be decrypted by authorized personnel.

Here is an example where we mask particular columns at source, encrypting them with AES.

Figure 12. A definition of a FEDSQL view with a masked column

The consumers of this view will see masked version of the PII which can only be reversed by using the decrypt function together with an associated key (that is, the key that was used for the encryption).

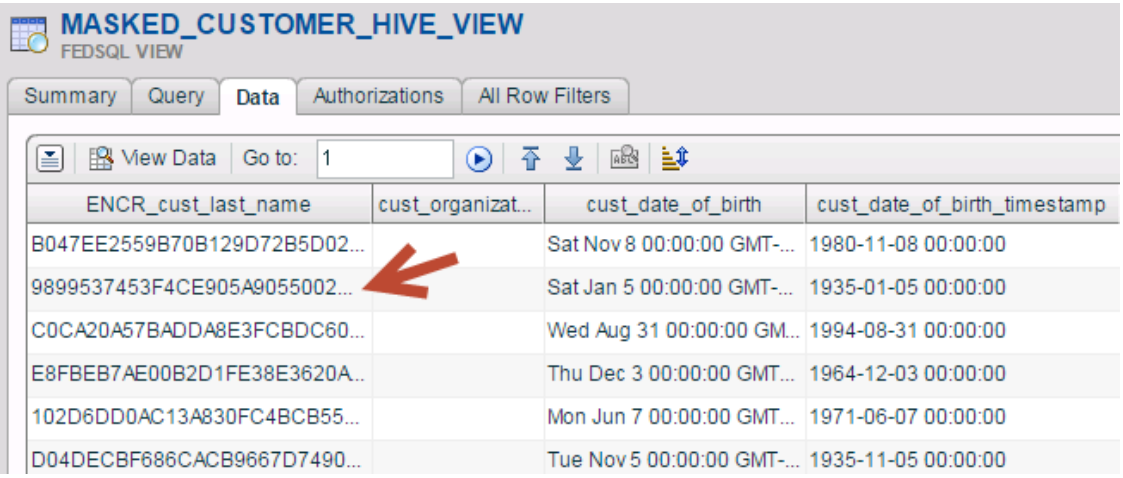


Figure 13. A masked (encrypted) column of a FEDSQL view

HOW CAN WE EXPLOIT THE DATA THAT SAS FEDERATION SERVER IS VIRTUALIZING?

SAS Federation Server can write result set data to SASHDAT. The SAS Federation Server Driver for SASHDAT is a write-only driver designed for use with Hadoop on the SAS LASR Analytic Server. The SAS LASR Analytic Server integrates with Hadoop by storing SAS data in the Hadoop Distributed File System.

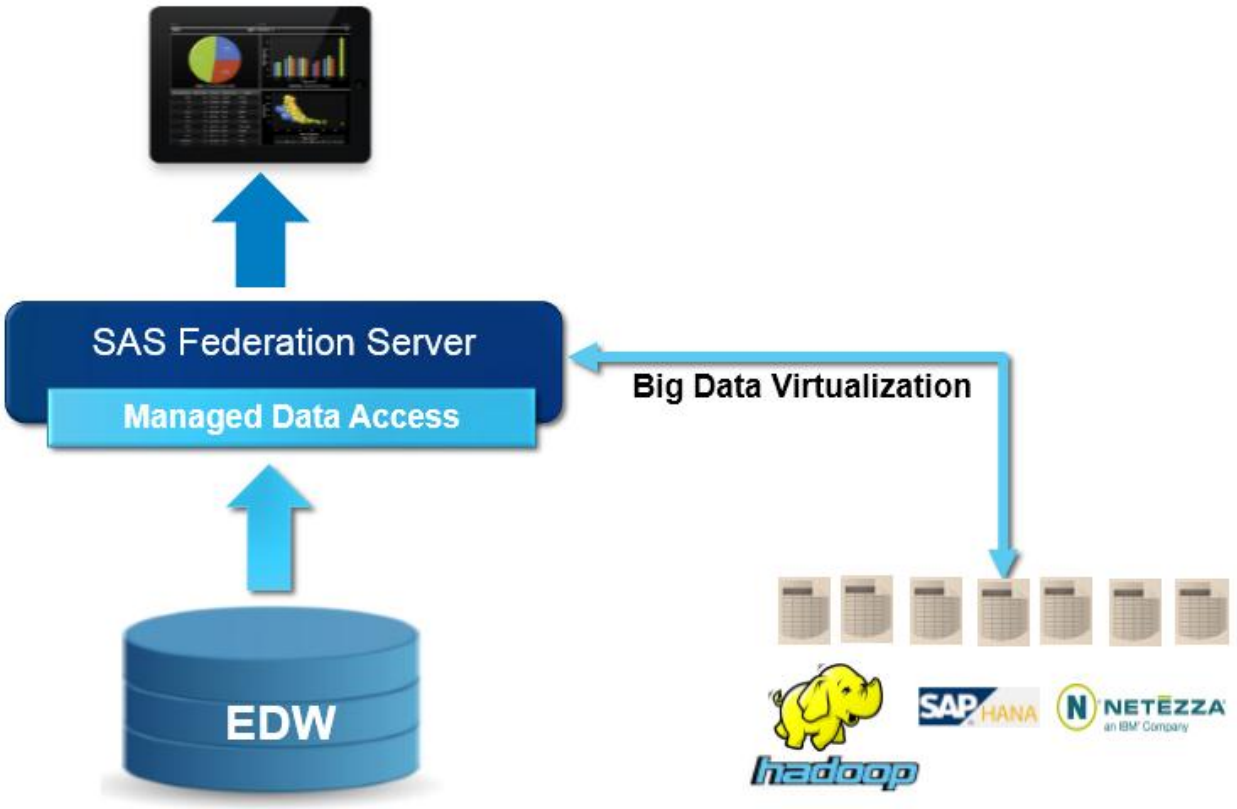


Figure 14. Delivering data for Data Visualization

We have already seen usage of SAS Federation Server managed data from SAS using the FEDSRV engine. This enables any SAS Solution to use Federation Server as a single-stop gateway to enterprise data.

for example,

```
LIBNAME FedHDP FEDSVR DSN=Hadoop PRESERVE_TAB_NAMES=YES catalog=HIVE
SERVER="sasbap" PORT=21032 USER="sasbap\sasdemo" PASSWORD="<password>";
```

SAS Federation Server is delivered with ODBC and JDBC drivers enabling comprehensive integration with third party applications – for example Microsoft Excel:

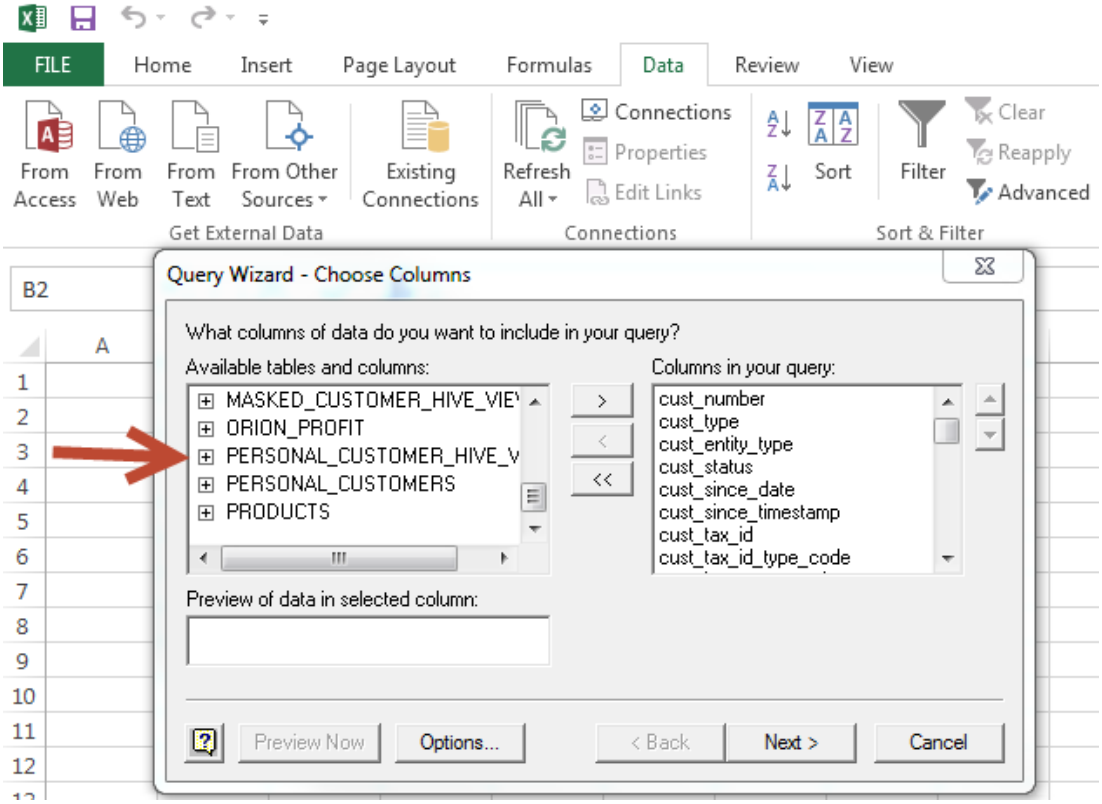
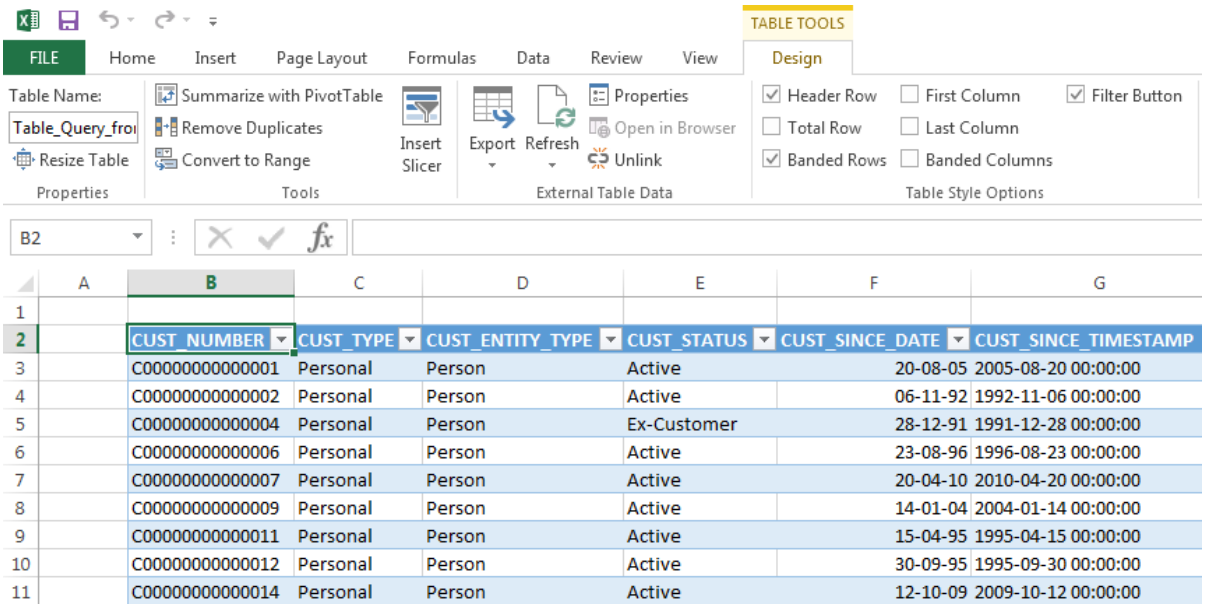


Figure 15. Accessing Hadoop data (FedSQL View) in Microsoft Excel



	A	B	C	D	E	F	G
1							
2		CUST_NUMBER	CUST_TYPE	CUST_ENTITY_TYPE	CUST_STATUS	CUST_SINCE_DATE	CUST_SINCE_TIMESTAMP
3		C00000000000001	Personal	Person	Active	20-08-05	2005-08-20 00:00:00
4		C00000000000002	Personal	Person	Active	06-11-92	1992-11-06 00:00:00
5		C00000000000004	Personal	Person	Ex-Customer	28-12-91	1991-12-28 00:00:00
6		C00000000000006	Personal	Person	Active	23-08-96	1996-08-23 00:00:00
7		C00000000000007	Personal	Person	Active	20-04-10	2010-04-20 00:00:00
8		C00000000000009	Personal	Person	Active	14-01-04	2004-01-14 00:00:00
9		C00000000000011	Personal	Person	Active	15-04-95	1995-04-15 00:00:00
10		C00000000000012	Personal	Person	Active	30-09-95	1995-09-30 00:00:00
11		C00000000000014	Personal	Person	Active	12-10-09	2009-10-12 00:00:00

Figure 16. Accessing Hadoop data (FedSQL View) in Microsoft Excel

Web Services (REST): The SAS Federation Server delivers an API that provides a simple method to access data. The API is implemented as a REST interface that provides direct interaction with SAS Federation Server, including metadata and data queries and SQL submissions. The SQL can include SAS Federation Server administration DDL or queries, DML, and DDL for back-end data sources:

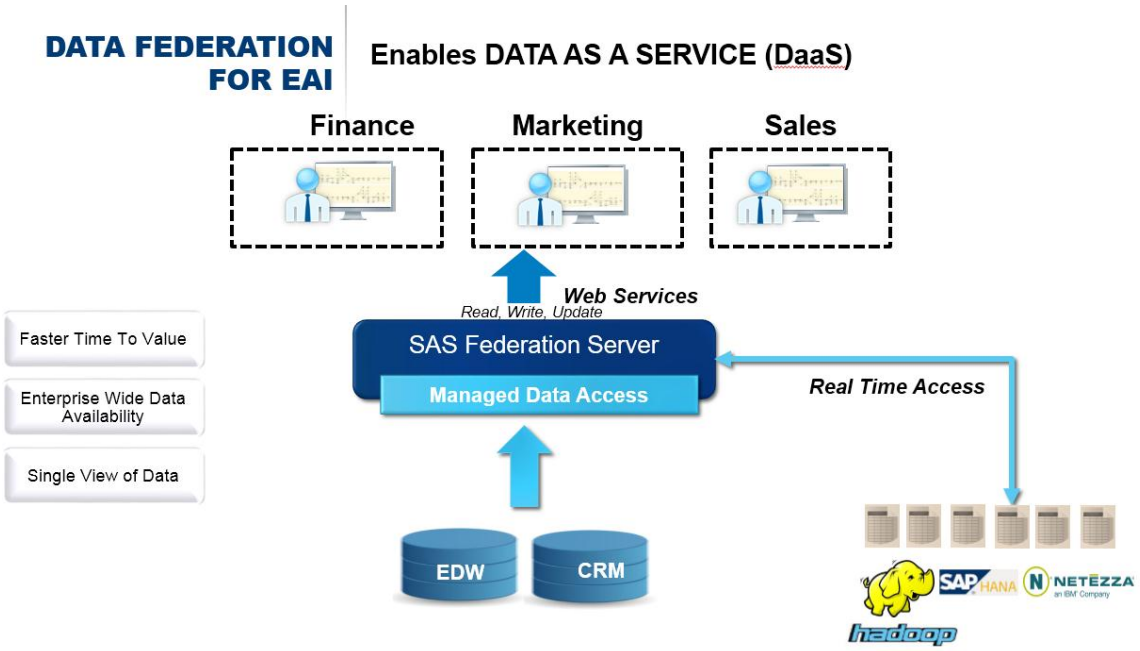


Figure 17. The SAS Federation Server REST interface (Web Services)

SAS Federation Server supports EAI (Enterprise Application Integration) by simplifying the data access via the web services. This makes data discovery even easier. Data services are exposed via a REST API for simpler access to data from 3rd party applications.

Faster Time to Value: Real time access to data via web services allows for data to be readily available for decision making purposes.

Enterprise Data Availability: The simplified web services data access approach allows us to access data that is available via the Web Services mechanism.

Single View of Data: Enables access to data from internal sources and external applications. This further helps in enabling a single of view of data sources for decision making purposes.

The following figures illustrate the development of Web Services queries, of our virtualized data, using the Dev HTTP Client plug-in for Google Chrome.

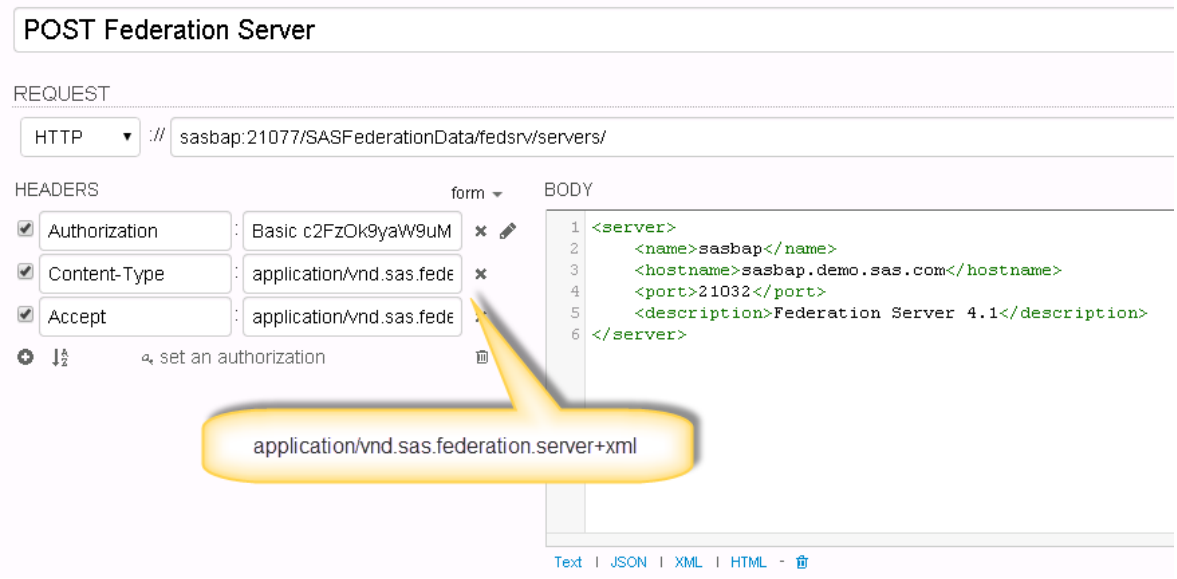


Figure 18. Using the Dev HTTP Client (for Google Chrome) to communicate with the Web Services

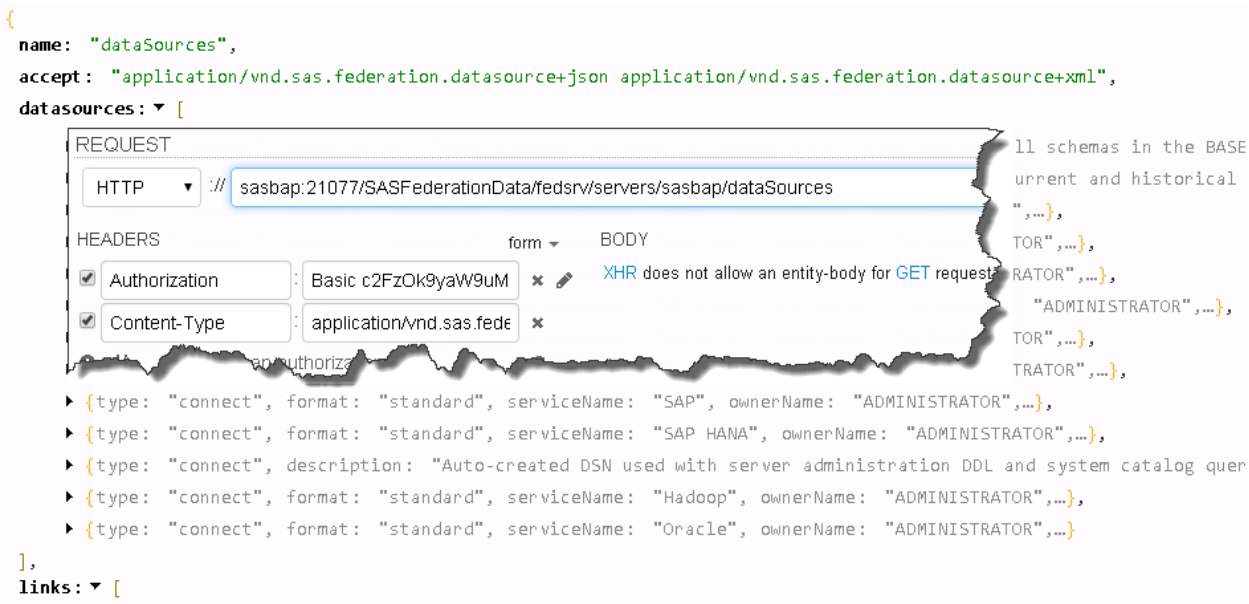


Figure 19. We can get a list of available Data Sources

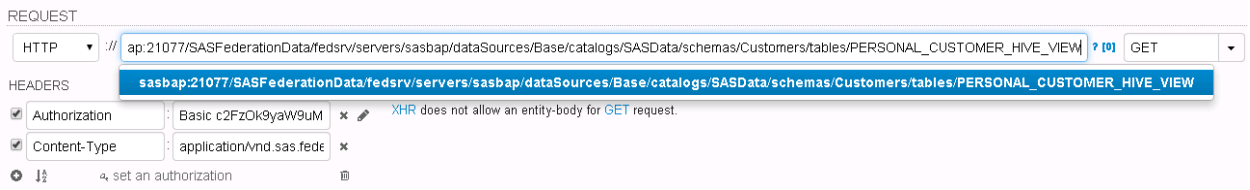


Figure 20. Ultimately we can get our data from the Federation Server cached FedSQL view

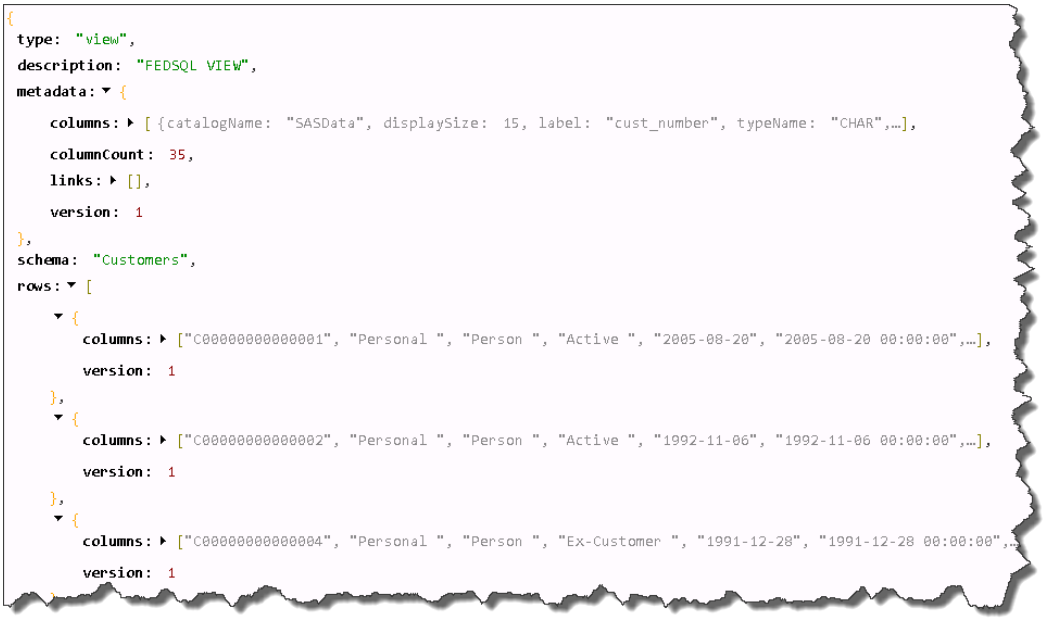


Figure 21. Data as a Service from a query executed in Hadoop HIVE

We can also construct cURL commands to perform the same queries to the Web Services. For example the previous query that returned data from our Federation Server cached FedSQL view could be issued as follows:

```
C:\>curl -i -X GET -H "Authorization:Basic c2FzOk9yaW9uMTIz" -H "Content-Type:application/vnd.sas.
federation.server+xml" http://race1:21077/SASFederationData/fedsrv/servers/sasbap/dataSources/Base
/catalogs/SASData/schemas/Customers/tables/PERSONAL_CUSTOMER_HIVE_VIEW
```

Figure 22. A cURL command to retrieve data from our Federation Server FedSQL view

Figure 23. The resultant data – which could be re-directed to a file

CONCLUSION

As we have seen, SAS Federation Server empowers business users to access and manage their own data. It provides a virtual layer or view, giving users the appropriate level of control without physically moving data.

We can access data, and that includes big data, in a consistent way across all the data sources. We can place administration and maintenance in the hands of IT experts while at the same time consumers of the data are enabled. The growing data security requirements and audit capabilities are served well by this single-stop shop for enterprise data.

We can harness big data sources and blend them with traditional systems in a seamless way. We can abstract technical environments and deliver a simple interface that is managed and monitored.

When we want to consume result data what better way is there than via web applications that interact with SAS Federation Server via the Web Services interface?

REFERENCES

SAS Federation Server: Administrator's Guide. Available at

http://support.sas.com/documentation/onlinedoc/fedserver/index.html

The SAS 9.4 FedSQL Language Reference, Third Edition. Available at

http://support.sas.com/documentation/onlinedoc/fedserver/index.html

The Dev HTTP Client for Google Chrome: RESTful API developer's Swiss knife. Information available at

https://plus.google.com/104025798250320128549/posts

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Ivor G. Moan
SAS Institute Inc.
In der Neckarhelle 162
69118 Heidelberg, Germany
Email: Ivor.Moan@sas.com
Web: www.sas.com