Paper 494-2013

IT in the sky: Building a private SAS® cloud

Andrew Macfarlane, SAS® EMEA Professional Services Delivery and Enablement
Frank Schneider, Allianz Managed Operations and Services SE, BI Shared Services

ABSTRACT

In today's climate, cloud computing is a defacto term used in IT and cloud capability a mandatory requirement for all software vendors. Based on real life experience, this paper will discuss challenges, opportunities and options for developing and implementing a private SAS® cloud using SAS® 9.3.

INTRODUCTION

To begin with, we should discuss what we mean by "Cloud".

As defined by the National Institute of Standards and Technology, Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.

For the purposes of this paper, we focus on some essential concepts of cloud computing:

1. Multi Tenancy (Resource pooling)
2. Scalability and rapid elasticity of resources
3. Shared Services

In addition, we dive deeper into the necessary building blocks of the Platform as a Service (PaaS) model.

PaaS describes the capability provided to the consumer to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

This all happen in a Private cloud. The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers (e.g., business units). It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises.

ECONOMIC ADVANTAGES

Economies of skill

In depth knowledge of the SAS Platform is a valuable but scare commodity. Comparatively speaking, it is expensive and hard to obtain those resource profiles.

By building a central managed platform under the supervision of a core nucleus of highly skilled employees, it assures lower costs, higher quality of SAS business services as well as a better match between availability and resources.

Economies of scale

The average cost for development of BI services provided by central service is lower than individually deployed BI projects. Costs are lowered by utilizing centrally located and maintained infrastructure and software to reduce up front set-up and on-going operating costs for each participating tenant.

Similarly In a cloud environment, the available resources may be better utilized (or sweated) through over commitment across all tenants. On a global shared platform, the workload pattern 'follows the sun' – whereby some customers are busy, whilst the others are asleep.

The higher utilization of the hardware results directly in lower infrastructure costs as hardware is never idle. Moreover, the necessary software licenses can be better used and the provider of the platform should be able to bundle purchasing power and negotiate lower prices.

Economies of scope

Building a platform based on standard, modular and reusable components provides the capability to leverage the effectiveness and efficiency by sharing best practices in terms of concepts, applications, infrastructure, and services among all tenants.

ARCHITECTURE VISION

For many enterprise level organizations that use SAS, especially those that have grown via mergers and acquisitions, Figure 1 describes a typical scenario of how SAS is deployed within such organizations. This scenario depicts local/department specific instances of SAS software, licensed and used for a specific purpose. Each environment is an independent silo.

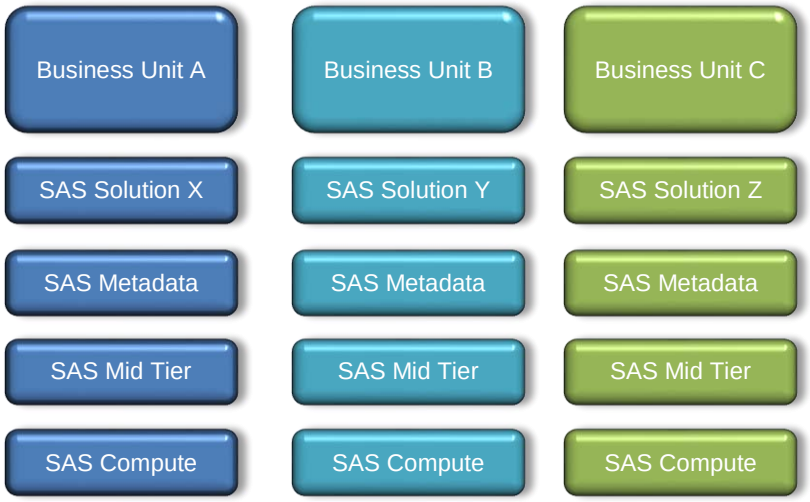

Figure 1 Silo'd local SAS deployments

This silo model leads to many challenges.

- Large volumes of duplication – in particular infrastructure and data.
- No common SAS maintenance strategy - each department is responsible for maintaining and managing their own environment(s).
- Expert Skills / or lack of, reside in each business unit.
- No consistent approach to running SAS across the organization.
- Lack of sharing of key business insight and value add.

To combat these challenges, the TO-BE architecture described in picture 2 could be designed. This model involves 2 levels of consolidation.

The first level of consolidation is through sharing ALL SAS Tiers between all business units. Secondly by deploying a variety of SAS products and Domain/Industry solutions onto these shared SAS tiers. All with the aim of provisioning a multi-tenant, multi-purpose, shared architecture through a common SAS Metadata Server, SAS Mid Tier and SAS Compute tier.

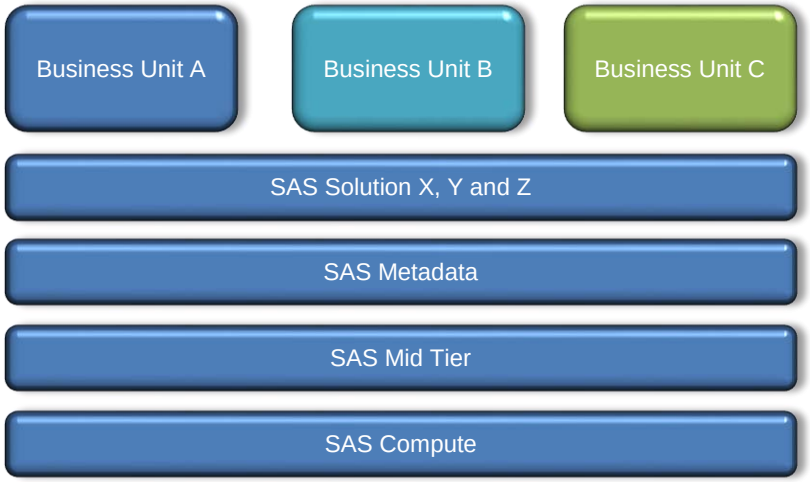

Figure 2 TO-BE Architecture - Shared SAS deployment

To design such an architecture, a large amount of effort is required to identify how the in scope SAS solutions could co-exist, highlighting and closing any gaps, and formulating an initial strategy for how each SAS Solution would peacefully co-exist beside others and the road map for delivery.

This can be particular challenging as due to the variety of SAS domain solutions, some may never have been originally designed for SaaS convergence.

ARCHITECTURE CONCEPTS

Multi Tenancy

Multi Tenancy is a key concept of cloud computing. Multi tenancy is also a fundamental enabler in releasing operational efficiencies in relation to support, maintenance and upgrades. For the purpose of our paper, we will use the following definitions for multi tenancy:

1. A single instance of an application shared across multiple organizations.
2. All applications must adhere to a common baseline of 3rd Party support software.

Details on third party support are available from the SAS support website:

http://support.sas.com/resources/thirdpartysupport/

From a core SAS (Business Intelligence/Data Management) perspective, there exist some common concepts to provide application Multi Tenancy.

One option for achieving multi tenancy could be through logical separation of tenants within the SAS Metadata Server and at the Operating System Level.

SAS Metadata provides many opportunities to provide a SAS container for a tenant. Consider how you could offer individual SAS Application Server Contexts, SAS Object Spawners, SAS Metadata Folders, Operating System Directories and system accounts that are unique to an individual business unit.

Each customer is granted their own SAS container, from where they have can use SAS as they wish.

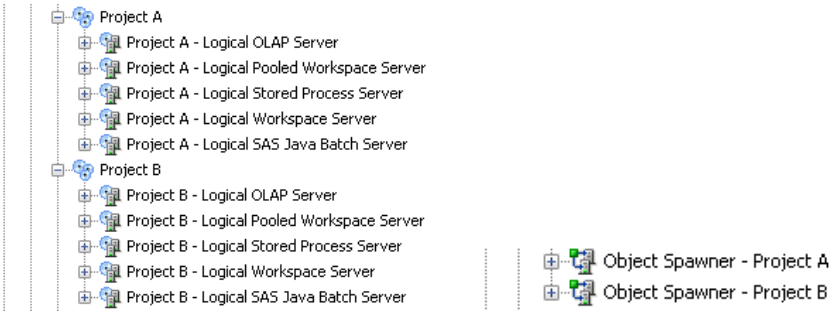

Figure 3 Potential SAS Metadata Components

For any SAS industry/domain solutions you wish to provision, you must dive deep into the internal workings of each solution, as there may be nuances in their design that do not fit your initial concept and architecture principles.

When implementing any multi tenancy concept, we need to keep in our thoughts how we can minimize the effort to on-board a new tenant.

SAS provides the Open Metadata API to allow scripted interactions with the SAS Metadata server. This API is a key enabler in automating processes that require SAS Metadata content to be generated, modified and also deleted if required.

Similarly, the SAS Platform object framework provides a series of pre-defined routines to support simple metadata interactions.

By combining these APIs with standard Operating System scripting techniques, you can rapidly decrease

a) The time to generate a new tenant

b) De-risk that process by having a robust, repeatable process, free of manual intervention.

Minimizing manual intervention is a key requirement for ensuring high system uptime.

SCALABILITY AND ELASTIC RESOURCE

The elasticity of resources is crucial in a private cloud. Elasticity is required to simplify the process of accommodating a new tenants' workload, or accommodating existing tenants who need more horsepower.

There are two general approaches to achieve this:

Scale vertically (scale up)

Scaling up is to add hardware resources (CPU/Memory) to a single node in a system. This option is available for every SAS layer (Web/SAS-Meta/SAS-App) within the platform.

As workload increases, you can add more resource.

Where this model can struggle is with the types of hardware required. A large symmetric multi-processor machine (SMP) such as an IBM P-Series or Oracle Sparc come with very high purchase costs, which is not ideal, especially if you do not require all resource in that server for your initial configuration.

Scale horizontally (scale out)

An alternative approach would be to scale out using clustering and load balancing techniques. In SAS 9.3, these techniques only apply to the SAS Mid and Compute tiers. *The only scaling option for SAS Metadata in 9.3 is a scale up model*. In a scale out model, we typically rely on commodity hardware (such as blades) and add more blades to accommodate increases in workload.

- SAS Compute Tier**:** Here, you could supply dedicated nodes per SAS application server context. If you need to scale a tenant beyond one node, you could also employ SAS proprietary load balancing (for Workspace, Stored Processes Servers, Pooled Workspace Servers and OLAP). In those configurations, you can simply add in more nodes as your workload increases.

  A final consideration is to utilize SAS Grid Manger for this scaling configuration. SAS Grid Manager provides additional load balancing intelligence (in place of round robin) to better utilize and distribute workload. SAS Grid Manager increases our ability to make sure we make best use of our resource pool at all times.

  Further details on load balancing in SAS can be found in the SAS(R) 9.3 Intelligence Platform: Application Server Administration Guide:

  http://support.sas.com/documentation/cdl/en/biasag/62612/HTML/default/viewer.htm#p07sasapplicserver000admingd.htm

- From a middle tier perspective, it is possible to increase the number of JVMs as the user workload requirements grow. Like the SAS compute tier, these JVMs can be provisioned across additional Mid Tier nodes. In this configuration, an HTTP-Server or Network switches can operate as the load balancer.

  In such a configuration, you need to consider that not all SAS web applications may be clusterable – for example the BI Dashboard event Generator and SAS Remote Services.

At this point, it is also worth considering your current and future storage requirements. Not specifically in terms of disk space, but more in terms of the technology available. Both SAN and Scale out NAS are ideal candidates for allowing future scaling of I/O throughput as your platform expands.

If you chose to use multiple SAS compute nodes and SAS datasets must be shared between those nodes, then a shared file system is a mandatory requirement (and also for SAS Grid).

Utilizing a shared file system can also open further avenues for system availability, as data (and therefore installation and configurations) can be made available to multiple servers, and avoid single points of failure.

AVAILABILITY

Compared to a local SAS deployment, running a private cloud brings many challenges from a system availability perspective. Firstly, there are now many different users, and voices on the system. Each group of users will have their own demands. When something impacts the system – such as it becomes unresponsive, no longer is the impact only to 1 division. It could be to every user of this cloud. To combat that threat, careful consideration must be given to providing resilience and increasing availability.

Some of the scaling techniques listed above serve to also increase the resilience of our cloud. Clustering and Load Balancing help support business continuity be having an N+1 architecture.

In the event of failure, there will likely be a smaller resource pool, however the system will continue to operate.

In order to reach the higher echelons of availability targets, SAS Grid Manager brings us some key capabilities. SAS Grid Manager can be configured to support high availability for any service that is critical to a SAS environment – such as the Metadata Server, Object Spawners. It can also control non SAS services too. This capability is provided by Enterprise Grid Orchestrator (EGO). EGO is a collection of cluster orchestration software components that, among other things, provide high availability to critical services. In the event of failure for a given service, EGO will assume responsibility for re-starting the failed process. Furthermore, should the server on which a critical service is running fail, EGO will launch that service on a subsequent SAS node.

By allowing EGO to control all of services, we can in practice have a series of SAS nodes with all components on each node – in an everything everywhere manner. In this configuration, any given node may perform any role in the SAS platform. Each node may be capable of being a Metadata Server, a Mid Tier node, or a Compute node. Using techniques such as DNS aliases or EGO Service Director to mask physical machine names within the SAS configuration, we can transparently move SAS services between any nodes we wish.

Having these capabilities is a key enabler in providing a highly available system.

Below is a typical high availability SAS deployment scenario:

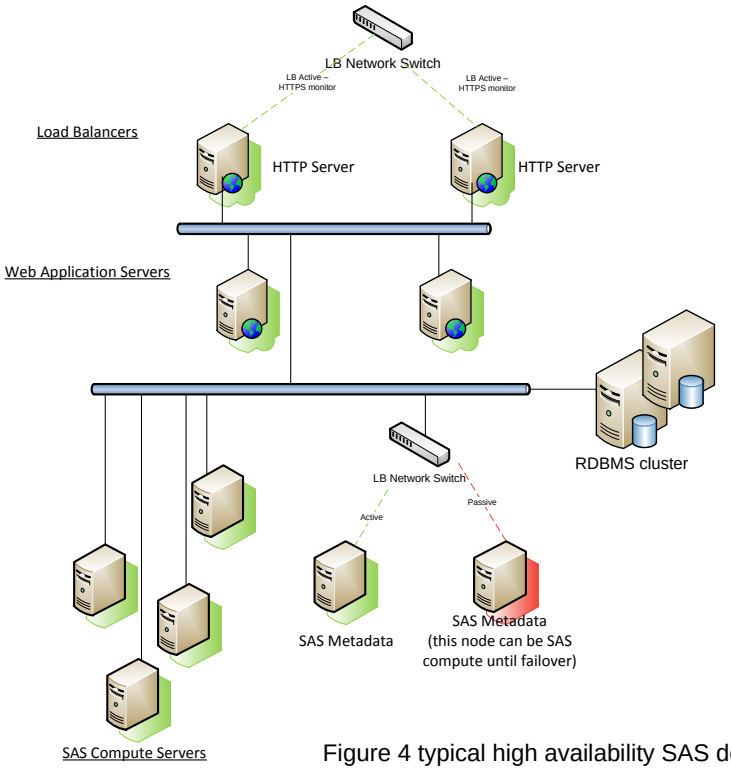

Figure 4 typical high availability SAS deployment scenario

Depending on your IT policies and standards, similar results can be achieved by SAS integration with other 3rd party high availability software such as IBM HACMP, HP Openview and others. Further details can be found here:

http://support.sas.com/rnd/emi/index.html

SERVICE MEASURMENT AND CHARGING

When providing a cloud, we must consider how to measure, and then charge our users. This can be an extremely difficult task, but it is crucial to ensure that your service is profitable. There are several options in these areas, such as:

Subscription based - charging a fixed fee over a fixed period.

Pay as you use – charge customer based on specific usage metrics.

When calculating these charges, you must consider the entire service you are offering – not just the hardware and software costs. What about support? How do you charge a customer for this? What about maintenance / service request, how may service requests do you provide as part of the cost, what scope can be covered by those service requests.

SAS Grid Manager can help in these areas though it's inbuilt accounting mechanism. This mechanism provides a method to capture exact resource usage, down to the user level.

Simpler methods can also be implemented through OS monitoring tools such as perfmon.

Whichever model you chose, you must be able to clearly articulate the usage figures back to your customer. Your customer will demand to know that they are being charged accurately for your service.

SECURITY

Even in a private cloud, your tenants will be keen to understand what security features you have enabled. In a cloud based environment, you no longer know who your near neighbour may be. To that end, security controls have to be implemented on all layers from the facilities (physical security), to the network infrastructure (e.g. end to end SSL communication for Web-Applications), to the IT system (Operating system, or Mid Tier Security), all the way to the information and application security–(SAS Metadata, and Data levels)

In these areas, the greatest asset is in expanding your knowledge of entry points into your cloud and how to secure them in a consistent manner.

One key example is the SAS General Servers account - or SASSRV as it is commonly called. This is a technical account which users inherit access to via SAS Metadata. It is imperative in a cloud environment that this account is not shared amongst all tenants. A more effective solution is to provide a dedicated SASSRV account for each tenant. That way, we can secure operating system directories and data even for the "on demand" SAS server processes that use these shared accounts.

IDENTITY MANAGEMENT

Identity Management within the enterprise is mainly focused on provisioning users and access policies. It is typically tightly integrated and directly connected to other enterprise systems – e.g. HR systems.
Due to the distributed nature of cloud computing the scenarios are a little bit different, so you have to consider some challenges[1]:

- How to synchronize identities between different enterprises and the cloud environment
- How to author/update/manage access policies in a manageable, low maintenance way

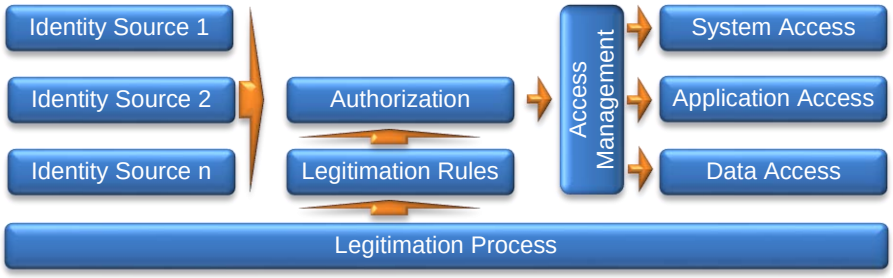

Figure 5 Identity and Access management Process

From a SAS Perspective you have to ensure consistent SAS metadata identities to assign Groups and Roles.

One way is to build up a bespoke identity management database with defined input interfaces (e.g. SAS BI Web Services) for identity sources, or other legitimation entitlement systems.

Where this is not possible a self-service interface could be a possible workaround allowing authorized users to manage this process. A disadvantage with this model is the requirement for one more administrative system interfaces for your customers.

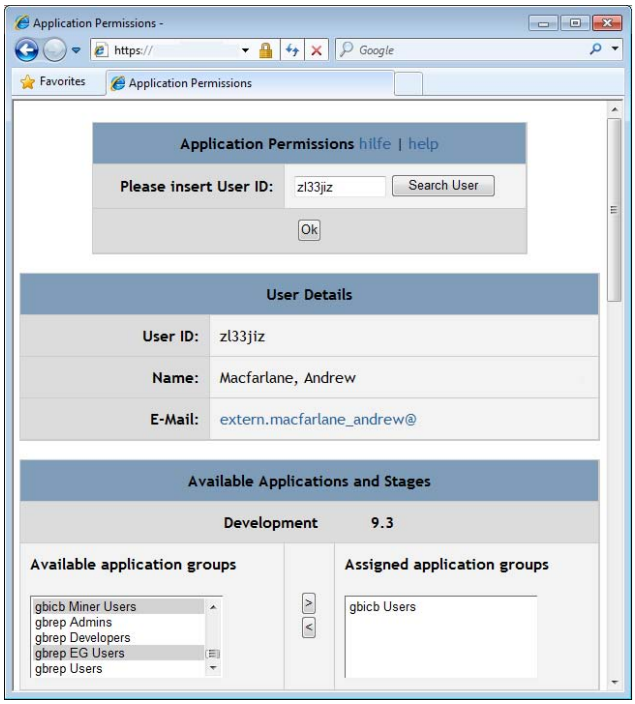

Figure 6 Example for a central access management system

[1] https://cloudsecurityalliance.org/guidance/csaguide.v3.0.pdf

SINGLE SIGN ON

A transparent Single Sign On is often a mandatory requirement for end users. As password management is not free, a SSO strategy helps to minimize operational costs.

But as many organizations have grown via mergers and acquisitions, often there is no homogenous IT-Landscape with a consolidated identity provider or directory. This is a challenge as the SAS-Login-Application is designed for only one authentication method.

Depending on the organisations IT-Landscape (heterogeneous), the Use-Cases (external identities,…) and the used components (Webserver, Web-Application-Sever) there is potentially a significant engineering effort necessary to deploy the authentication methods that fits to the organisation.

For example, that engineering may take the form of custom login modules to facilitate multi-level authentication methods:

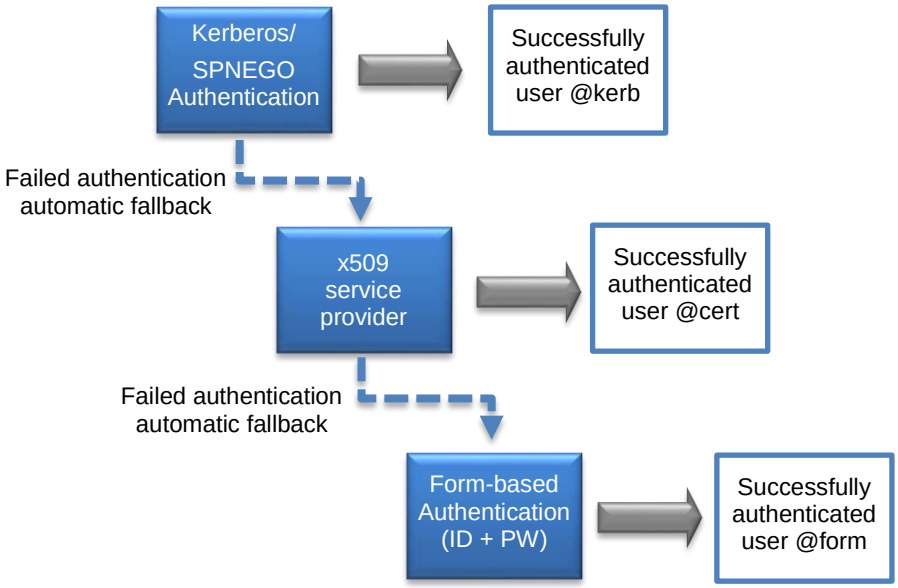

Figure 7 Example for Login-Workflow

CLIENT DEPLOYMENT

Deployment of the SAS Clients is one of the more challenging prospects in the design of a private SAS-Cloud. To cater for a wide distribution (geographically) and also accommodate any forward/backward compatibility between SAS Client releases, and SAS Server releases.

Forward and Backward compatibility is vital in reducing the number of clients you need to maintain and it removes the need to update clients every time a new version is released. If we think about the operation of a private cloud, no longer do we have 1 community to consider when we need to upgrade clients, we need everyone's agreement.

Brining new clients online can have a negative impact on support – end users require new training, sometime new clients prompt more service calls until users become accustomed to their features. Clients such as SAS enterprise Guide and the SAS Add-in for Microsoft Office generally perform well in this area. If you use JAVA based clients such as SAS DI studio, then you will need to consider how you can support multiple releases simultaneously.

To distribute over a wide geographical area, local desktop deployment is not an option. There options are:

1. To package and distribute the SAS Desktop clients via software packaging and publishing tools.
2. Terminal servers – to deploy a single instance of SAS clients that can be accessed by many users.
3. Application virtualization - including streaming

From SAS 9.3 onwards, SAS provides (on a reasonable effort basis) assistance to customers who wish to provide desktop applications through streaming technologies.

Further details can be found here:

http://support.sas.com/techsup/pcn/virtualization.html

LOCALIZATION / UNICODE

To enable a global reach of cloud services, UNICODE SAS and RDBMS installations are necessary. Utilizing UTF-8 bypasses issues of local encoding, and supports processing all customer data from any locale from the same SAS invocation and then store the outputs in a consolidated manner.

SAS provides dedicated "K" character functions to accommodate double byte character data. When using UTF sources, you must ensure that all development (including SAS Solutions) use these functions. Otherwise you may end up with unexpected results.

Likewise consideration needs to be given for any additional disk space required to store these double byte character sets.

WATCH THIS SPACE

SAS is committed to furthering its cloud strategy. SAS is fostering on-going actions that will expand their software as a service offerings, and also introduce new platform as a service offerings.

SAS 9.4 will bring further technology enhancements to help enable cloud deployment, such as SAS IDE and SDKs, automated maintenance; likewise there exists already several SAS customers running on AWS. All of which points to a bright future for SAS in the cloud.

SUMMARY

With cloud computing being an industry expectation, this paper addresses how, with some forward planning and detailed knowledge of SAS Software, we are able to deploy SAS software in a manner that fulfils the key capabilities of a cloud – specifically Multi Tenancy, Elastic Resource and Shared Services.

Building such a platform is a long term strategy that requires many transitional architectures until the final goals are reached. Many of the individual topics discussed in this paper are themselves common deployment scenarios for SAS customers. SAS customer successes demonstrate that through bringing all of these deployment scenarios together on one system, a private SAS cloud is more than just a dream.

REFERENCES

http://csrc.nist.gov/publications/PubsSPs.html#800-145

http://www.gartner.com/DisplayDocument?doc_cd=225922&ref=g_noreg

Cloud Charging Models

http://www.cisco.com/en/US/services/ps2961/ps10364/ps10370/ps11104/Cloud_Services_Chargeback_Models_White_Paper.pdf

SAS Open Metadata Interface

http://support.sas.com/documentation/cdl/en/omaref/63063/HTML/default/viewer.htm#omarefwhatsnew93.htm

CONTACT INFORMATION

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

Andrew Macfarlane

SAS EMEA
Professional Services Delivery and Enablement

andrew.macfarlane@sas.com

Frank Schneider

Allianz Managed Operations and Services SE
BI Shared Services / Global BI Platform

frank.schneider_hv@allianz.de