OPTIONS FOR MIGRATING TO ALTAIR SLC AND THE CLOUD

Introduction

Many corporations, government agencies, and other large organizations use the SAS language to develop and deploy data management, data preparation, extract-transform-load (ETL), advanced analytics, predictive modeling, multivariate analysis, business intelligence (BI), reporting, and other applications. Until recently, modernizing a SAS language software environment was synonymous with rewriting all applicable applications into another language.

Altair SLC™ provides a new option: users can switch suppliers in the same way as they might select a different C++ or Java compiler or a different database vendor while retaining their investment in existing IP and applications. This transforms the outlook for SAS language users seeking to build a future-proof data analytics strategy. Organizations can migrate to Altair SLC, which runs those same SAS language applications at reduced cost, along with the flexibility to embrace Python, R, and other technologies, including cloud platforms.

This guide lays out the options for anyone considering a move to Altair SLC and/or the cloud, and the factors to consider when making such a strategic decision.

Typical Deployment Scenarios

Daily operations for many organizations using SAS language depend on applications they have developed using the language. These programs support broad and varied business needs, including credit scoring, marketing analytics, churn analysis, fraud detection, insurance pricing, risk assessment, financial compliance, clinical research, and many other data-centric business processes.

Most organizations that have written SAS language programs have deployed them on-premises, often utilizing mainframe computers, servers, and workstations. Many are planning to modernize their infrastructure by moving to cloud and/or hybrid cloud/on-premises deployments; some are concluding that staying on-premises is best for them.

Reasons to Move

Every organization and situation are unique, but common drivers for a migration include:

- Reducing total cost of ownership (TCO)
- Obtaining more flexible licensing terms
- Tapping internal and external skills pools familiar with Python and R, and visual, code-optional data science tools
- Adopting new cloud-native facilities and technologies
- Product support lifecycle mandates replacing current product/platform

Major Options for Modernized Deployments

A detailed examination of the pros and cons for each option is beyond the scope of this paper, but this section provides a high-level look at various strategies available to an enterprise.

	Stay On-Premises Existing Vendor	Stay On-Premises Altair SLC	Move to Cloud Existing Vendor	Move to Cloud Altair SLC	Rearchitect On-Premises Open Source	Rearchitect to Cloud Open Source / Cloud-Native	Rearchitect To Cloud Existing Vendor
Regulation impedes rearchitecting	●	●	●	●			
Cloud functionality inadequate	●	●					
Cloud performance inadequate	●	●					
High cloud migration and operational costs	●	●					
High cost to migrate complex applications	●	●	●	●			
Unsuitable functional results from cloud platform	●	●	●				
Anticipated lifetime of applications is short	●	●	●	●			
Product support lifecycle mandates change		●		●	●	●	
Limited business motivation for change	●	●	●	●			
Enable Python, R, SQL		●			●	●	●

	Stay On-Premises Existing Vendor	Stay On-Premises Altair SLC	Move to Cloud Existing Vendor	Move to Cloud Altair SLC	Rearchitect On-Premises Open Source	Rearchitect to Cloud Open Source / Cloud-Native	Rearchitect To Cloud Existing Vendor
Vendor assistance moving to Python, R, SQL		●		●			
Budgetary pressure		●		●			
Strained vendor relationship		●		●	●	●	
Flexible licensing terms		●		●	●	●	
Rush to cloud			●	●			
Long-term strategy to cloud				●			●
Low complexity applications	●	●	●	●	●	●	●
Open cloud application context				●		●	

Stay On Premises — Existing Vendor

This strategy calls for the organization to retain its on-premises infrastructure, retain all applications unchanged, and retain SAS Institute® as a software vendor. This option offers very low migration cost, high operating cost, potentially high capital costs in the future, and relatively low risk.

Stay On Premises — Altair SLC

This low-risk approach supports low migration costs and ongoing costs since it retains on-premises infrastructure with unaltered user-written application programs. This approach has the potential to reduce costs and supports more flexible licensing terms. Altair SLC also provides high performance, reliable management and deployment tools, and allows organizations to continue utilizing the skills of their trained analytics staff. Altair SLC Hub uses a Go-based microservice architecture open to Python, R, SQL, and SAS language coding.

Move to Cloud — Existing Vendor

This low-risk approach results in a cloud-based infrastructure with largely unaltered, user-written applications and continued licensing from the legacy software vendor. The strategy supports relatively low migration costs (dependent on usual on-premises to cloud transformation issues) but does not escape potentially inflexible licensing terms coupled with associated high ongoing software licensing and new cloud-infrastructure costs.

Move to Cloud — Altair SLC

This approach offers low risk, low migration costs, low ongoing costs (albeit combined with new cloud-infrastructure costs), enables retention of existing user-written applications, and supports flexible licensing terms and options for further modernization. Altair SLC also provides high performance, reliable management, and deployment tools, and allows organizations to continue utilizing the skills of their trained analytics staff. Altair SLC Hub uses a Go-based microservice architecture open to Python, R, SQL, and SAS language coding.

Rearchitect On Premises — Open Source

There are many options for rearchitecting user-written applications on-premises with some options creating more disruption and more extensive rewrites than others. The major cost of rearchitecting is typically testing and QA in addition to the actual redevelopment of existing applications in a new language. Managers must consider that older applications may have no specifications, user documentation, and/or documented test routines.

This high-risk approach brings high migration costs and unknown ongoing costs. Open-source software can be free but is often supported by expensive maintenance and service agreements.

Popular target environments include Python, R, and SQL. These popular languages offer large developer recruitment pools. However, the cost of rewriting and testing applications is high and difficult to estimate accurately before launching a project. The rewritten application code will typically be more verbose and offer lower levels of performance compared to the original code written in the highly efficient SAS language. Operating costs are difficult to estimate as well, making this a high-risk and potentially high-cost solution.

Rearchitect to Cloud — Open Source / Cloud-Native

There are additional options for rearchitecting applications to the cloud with some options creating more disruption and requiring more extensive rewrites than others. Moving to the cloud enables organizations to incorporate additional cloud-native services into their target solution. The organization must select a specific cloud vendor, at which point the solution is likely to become, to some extent, proprietary.

The major cost of rearchitecting to the cloud is typically testing and QA in addition to the actual redevelopment of existing applications in a new language. Managers must consider that older applications may have no specifications, user documentation, and/or documented test routines.

This high-risk approach brings high migration costs and unknown ongoing costs. Open-source software can be free but is often supported by expensive maintenance and service agreements. Cloud service fees will increase operating costs.

Popular target environments include Python, R, and SQL. These popular languages offer large developer recruitment pools. However, the cost of rewriting and testing applications is high and difficult to estimate before launching a project. The rewritten application code will typically be more verbose and offer lower levels of performance compared to the original code written in the highly efficient SAS language. Operating costs are difficult to estimate as well, making this a high-risk and potentially high-cost solution.

Rearchitect to Cloud — Existing Vendor

This approach is essentially rearchitecting as well. Migration costs are likely to be high combined with the risks associated with significant technology differences between the legacy platform and the cloud platform. This approach also results in an even stronger vendor-lock.

SAS Viya® provides different functions for solving problems compared to the SAS language. Migration may require significant rewrites of applications. SAS Viya® is proprietary, and for many SAS language applications, SAS Viya® may not be a suitable replacement. SAS Viya® is typically substantially more expensive to license than traditional SAS Institute® software products. This high-risk approach also brings high operating and capital costs, since it typically requires a large compute environment.

Selecting Modernization Strategies

Many global enterprises using the SAS language have very large numbers of programs and models (running into the millions in some cases) deployed in various ways, fulfilling many different business-critical requirements. As such, it is likely that no single option will be suitable for a comprehensive SAS language modernization strategy. Large organizations may consider multiple options when modernizing depending on:

- Regulatory constraints
- Tolerance of business function to risk
- Performance expectations and SLAs
- Costs, budgets, and timeframes
- Ease of adoption, training and resourcing of administrators, developers, and end users
- Desire to introduce new capabilities and facilities
- Strategic imperatives

People

Users have varying degrees of willingness and ability to accept new technology, regardless of how similar it is to what they are familiar with. There are likely to be significant training requirements associated with moving to a new language or a new user interface. Administrators, power users, and IT teams will also need training on the modernized system. Moving to Altair SLC requires minimal retraining and minimizes expenses, disruptions, and the risk of error compared to other technologies.

Process

Big organizations tend to support large numbers of business workflows using SAS language applications, many of which are unique. Critical processes (for example, automated credit risk decisions) are often subject to multiple levels of government and/or industry regulation. Changing technology can require expensive and time-consuming re-validation. To remain compliant, models and decision-making processes must be transparent and explainable to regulators. The SAS language is widely understood and trusted by regulators.

Technology

Core technical capabilities provided via SAS language include access to and consumption of data and development and execution of models that produce recommendations and/or decisions. The organization must maintain high levels of reliability, resilience, scalability, elasticity, performance, and cost-effectiveness. Altair SLC is highly performant; it consumes limited amounts of resources compared to other technologies that solve performance problems by utilizing very large-scale computing resources.

Typical Uses of SAS Language Applications

When determining how to modernize an SAS language estate, it is important to understand how SAS language capabilities are being used and what an effective modernization strategy should consist of for each scenario.

Organizations use the SAS language in a wide variety of applications and business areas, including:

- Data management
- Data preparation
- Extract-transform-load (ETL)
- Advanced analytics
- Predictive modeling
- Multivariate analysis
- Business intelligence (BI)
- Reporting

SAS language applications often leverage the full range of the SAS language's powerful capabilities to support:

- Market research
- Credit scoring
- Churn analysis
- Fraud detection
- Insurance pricing
- Financial compliance
- Risk management
- Decision support
- Clinical research
- Healthcare
- Manufacturing

Key Technology Considerations

Data Storage

Proximity of data stored locally, remotely, distributed and in the cloud is a key factor in runtime performance. For example, persistent data files may be stored in departmental file shares or in cloud storage buckets. Temporary and work files used when running SAS language programs, however, require fast local storage.

Data may be stored in spreadsheets, departmental databases, enterprise databases, onsite data warehouses, cloud databases, and big data repositories. Data processing is ideally located near to where data is stored, and sometimes pushed into the data store itself. Where data is distributed, it is often valuable to be able to split and distribute processing. The SAS language and Altair SLC support distributed and pushed-down processing of data using a variety of configurations to suit different scenarios.

Service Level Agreements (SLAs)

It is important to understand SLAs between stakeholders including business users, IT, external customers, partners, and regulators. Business users may require and expect that machine learning models produce results within a specific timeframe. Regulators may expect a specific format of output. Customers may have paid for regular delivery of certain artifacts or APIs.

SAS Language Performance Requirements

Properly built and deployed SAS language applications are fast and efficient. Implementation teams must carefully consider throughput and performance characteristics and understand the costs of target technologies, infrastructure, and architecture. Such requirements can be highly variable depending on the specifics of each deployment and workload (for example, basic queries versus advanced analytics.) Altair SLC reduces the likelihood of degraded outcomes compared to other technologies.

Licensing and Environment

Licensing can become complicated when moving SAS language workloads to the cloud:

- Existing restrictive license terms (in particular, between on-premises and cloud deployments and between geographic regions)
- Pending obligations to upgrade or re-configure to be compatible with new products
- Flexibility of Altair licensing terms including Altair Units
- Estimating data volumes, network ingress/egress, and processing capacity

Language Modernization Pathways

IaaS versus PaaS

Modernization planning teams may consider utilizing:

- Infrastructure-as-a-service (IaaS)
- Platform-as-a-service (PaaS)

The PaaS option tends to bring higher technical and management migration complexity, expense, and risk than the IaaS option. Ongoing PaaS operations can, in some cases, be simpler in the long term as platform providers take responsibility for more of the technology stack. IaaS can give organizations greater control over set-up, management, and security. Altair SLC is often targeted at an IaaS infrastructure, although Altair also offers a PaaS solution.

An organization could choose, for example:

- IaaS for SAS language applications using the required number of server instances or Kubernetes
- PaaS with more fundamental changes to architecture and re-platforming of SAS language applications to make use of cloud-native services

Python and/or R and/or SQL

Many enterprises are debating the use of Python or R as a target replacement programming language for at least some SAS language applications. Altair is committed to supporting Python, R, SQL, and SAS language applications fully. Altair SLC makes it as easy as possible for users to switch from one language to another and to convert programs from one language to another.

Altair SLC products enable users to develop applications incorporating all four languages together, including bi-directional interoperation. For example, Python and SAS language teams can develop and run applications in either language and work together to make full use of existing and new IP. Altair SLC also includes features to assist conversion of SAS language application code into Python, R, or SQL code.

Altair's Unique Units Licensing System

Altair Units, Altair's patented units-based licensing system, is scalable and shareable across an organization. It allows customers to maximize software licenses through the flexibility to run Altair software anywhere, and the freedom to choose from a variety of software tools with unparalleled value.

With Altair Units, customers have full access to Altair software instantly, including more than 150 Altair and partner products, and can run these applications on-demand locally, in the cloud, or in a hybrid setup. Learn more about Altair Units.

The Altair RapidMiner Platform: A Comprehensive Range of Data Analytics Tools

Altair SLC and related products are part of Altair RapidMiner, our data analytics and AI platform.

- Artificial Intelligence and Machine Learning: Altair's industry-leading visual approach to analytics modeling minimizes repetitive tasks, shares knowledge across enterprises, and reuses steps within connected model workflows for faster analysis and shared insight. Altair tools support code written in the SAS language, Python, R, and SQL and offer a unique visual interface for creating machine learning models without requiring any written code.
- Data Preparation: Access, cleanse, and format data from a wide variety of sources (including Microsoft® Excel®, CSV, PDF, TXT, JSON, XML, HTML, SQL databases, big data sources like Hadoop, and more) without any manual data entry or coding. Automate data transformation and report distribution tasks. Integrate data transformation workflows with enterprise content management (ECM) and robotic processing automation (RPA) platforms.
- Business Intelligence and Streaming Data Visualization: Build, modify, and deploy sophisticated business intelligence, data visualization, and stream processing applications using a drag-and-drop interface. Connect to virtually any data source, including big data sources, SQL and NoSQL databases, message queues; develop complex event processing programs; and design visual user interfaces that provide the perspectives people need to make insightful, fully informed decisions based on massive amounts of data.

SAS Institute and SAS Viya are trademarks of SAS Institute Inc.