Astadia

THE MAINFRAME TO CLOUD EXPERTS

Unisys to Azure Reference Architecture

Astadia Mainframe-to-Cloud Modernization Series

Abstract

In businesses today, across all market segments, cloud computing has become the focus of current and future technology needs for the enterprise. The cloud offers compelling economics, the latest technologies and platforms, and the agility to adapt your information systems quickly and efficiently. However, many large organizations are burdened by much older, previous generation platforms, typically in the form of a mainframe computing environment.

Although old and very expensive to maintain, the mainframe platform continues to run the most important information systems of an organization. The purpose of this reference architecture is to assist business and IT professionals as they prepare plans and project teams to start the process of moving mainframe-based application portfolios to Microsoft Azure. We will also share various techniques and methodologies that may be used in forming a complete and effective Legacy Modernization plan.

In this document, we will explore:
- Why modernize a mainframe
- The challenges associated with mainframe modernization
- An overview of the Unisys mainframe
- The Unisys to Azure Reference Architecture
- An overview of Azure services
- A look at the Astadia Success Methodology

This document is part of the *Astadia Mainframe to Cloud Modernization Series* that leverages Astadia's 25+ years of mainframe platform modernization expertise.

Astadia — THE MAINFRAME TO CLOUD EXPERTS

Contents

This page intentionally left blank.

Introduction

The Microsoft Azure cloud computing platform is an excellent target environment for transitioning from a mainframe workload to a cloud implementation. With the security features of Azure and the ability to scale based on demand for the services, Azure offers a complete operational environment in support of mainframe workloads that have been migrated to the cloud. In addition, Azure supports innovation of the application portfolio, previously held captive by the inflexible nature of a mainframe computing model, improving the productivity of application developers and support personnel.

Even more than a typical IT project, planning to modernize mainframe applications is the most important phase of the total project effort. A good place to begin is with a thorough assessment of the existing overall mainframe application portfolio. Through the assessment process, all aspects of the existing portfolio will be inventoried and examined in detail, resulting in a catalog of each application, database, technology platform and business user profile currently in use. Once completed, the results of this application rationalization will then guide the sequence of application migration, as well as the different modernization strategies and techniques that may be called upon over the course of the entire project. We've included an overview of how Astadia tackles Legacy Modernization projects with our Success Methodology to give you an idea of what's involved.

Don't let the enormity and importance of a mainframe modernization project deter you from getting started. The skilled individuals needed to continue to maintain mainframes are increasingly leaving the workforce through retirement and are not being replaced. Hardware and software maintenance costs continue to escalate and the demands of customers, employees and partners require greater innovation than mainframe platforms can support.

How to Use This Reference Architecture

Begin by reading the "Why Should I Migrate" section first. From there:

Mainframe Experts: Skip to the Unisys to Azure Reference Architecture and the Understanding Azure sections

Azure Experts: Start with the Understanding Unisys section, followed by the Reference Architecture.

Business Leaders: Spend time with the "Why Should I Migrate…" section and the Ensuring Project Success section at the end.

About Astadia

Astadia is a Microsoft Gold Partner that has been in the legacy modernization business since 1994 and has successfully completed more than 200 mainframe modernization projects.

Our repeated success has allowed us to develop a comprehensive methodology, proprietary software tools and techniques, as well as the "know how" that comes with more than 25 years of experience handling mission critical applications and data.

We're pleased to share some of that experience with you through our Mainframe to Cloud Modernization Series of reference architectures, webinars, whitepapers and more.

Visit our website at www.astadia.com for additional information.

Why Should We Migrate Our Mainframe Apps to Azure?

Over the past 10 years, public cloud computing has emerged as the foundation of future enterprise technology. In terms of technology generations, mainframes are at least two generations old, perhaps three. Yet, they still survive today and are responsible for running key financial, healthcare and other vital and sensitive systems around the world.

So, why should you migrate your mainframe workloads, why migrate them to Azure and why is now the right time?

Benefits of Mainframe Modernization

The specific benefits in moving any mainframe workload will vary between organizations and even at the application and database level. In general, here are three of the top reasons driving legacy modernization projects today:

Cost – The economics of cloud computing are compelling when compared with the status quo of maintaining a mainframe environment. A total cost of ownership (TCO) evaluation of the subscription-based, consumption driven cost model of the cloud versus the exorbitant hardware & software maintenance costs of mainframes will show a very appealing and short-term achievable ROI (potentially less than 12 months from project completion).

People – Mainframe-specific technical skills are not being replaced by today's college or technology trade school graduates. The pool of available talent with relevant knowledge and experience is shrinking exponentially each year. The cloud leverages modern technology and its use is ingrained into young software engineers worldwide.

Flexibility – The cloud offers an Open Systems environment in which high productivity and rapid innovation happen at a tremendous rate. A properly designed implementation of a cloud infrastructure scales easily and quickly, both expanding and collapsing to synchronize with business demand. Backup, redundancy and disaster recovery is seamless. Support for multiple end-user platforms and devices is inherent. Database sharing across the enterprise with high performance is achievable.

Approaches to Mainframe Modernization

You may notice throughout this document that we use the terms "Mainframe Modernization" and "Mainframe Migration". Migration is a type of modernization, whereas modernization encompasses a broader set of strategies or options. In many cases, you will employ a combination of these strategies, the right mix of which ones will be determined during the critical application portfolio rationalization step of the project's assessment phase. Here are three of the most common approaches:

Reuse – Often called "lift and shift", this is a process that reuses the existing code/program/applications, typically written in COBOL, by moving them off the mainframe, and recompiling the code to run in a mainframe emulator hosted in a cloud instance. This approach minimizes the upfront risks and the length of the project, realizing hardware and software cost savings soonest.

Running mainframe applications in an Azure-hosted emulator also opens the possibility of innovation leveraging APIs to previously inaccessible programs and data.

Rewrite – It may be tempting to say, "Let's just write new programs from scratch," to modernize the mainframe applications. This approach is extremely risky and fails a vast majority of the time. It is complex, costly, and time consuming. The resources and investment required tends to greatly exceed the forecast.

A new, modern codebase may still be the correct end objective, but a better approach would be to first move the applications to a cloud-based emulator, migrate the database to a cloud-based database, then focus on replacing modules/code over a deliberate, multi-phased approach. When it is time to rewrite, there are several code transformation engines you can choose from to reduce the effort and minimize the risk.

Replace – Another mainframe modernization approach is to completely replace the mainframe functionality with a program or suite of programs, typically a Software-as-a-Service (SaaS) application. You typically see this with purpose-built solutions for finance, human resources, manufacturing, enterprise resource planning, etc. There are also industry specific apps that may solve the problem that a custom mainframe solution was needed for decades ago.

The upside of using SaaS is that your organization no longer worries about maintaining code. However, you will find that while you can *configure* a SaaS application with various options provided by the vendor, you will not be able to *customize* your instance, as the shared codebase runs all tenants (customers/organizations) using the "service".

There are additional variations on these three modernization approaches and you'll likely use several strategies in achieving your goal to completely migrate from the mainframe. It is commonly accepted best practice among legacy modernization practitioners to primarily use the lower-risk, lower-cost Reuse approach first to capture the gains and benefits in the shortest time possible, followed by a deliberate and phased approach to Rewrite or Replace the applications.

Challenges of Mainframe Modernization

Mainframe migration projects are complex and require close management of the process, budgets and timelines that have been set as project goals. A Reuse approach will involve rehosting (from mainframe to Azure) and likely some re-engineering and refactoring to complete an entire mainframe migration. It will also involve data and file conversions for transitioning the database to the cloud.

As we've been emphasizing, the first challenge of any mainframe modernization project is to develop a rock-solid plan built upon a thorough application portfolio assessment and rationalization. As you put your plan together and begin to execute, here are additional factors you'll need to watch out for:

Documentation

Many mainframe environments with large and complex application portfolios do not have documentation that details what these mainframe applications do, and how they do it. Many applications are decades old, so the original system, with changes likely every year, has become a maintenance nightmare. The external interaction with these systems, the Input/Output, is how these systems get defined to the business, and the rest of the system is just a black box.

Migrating a minimally-documented system of this nature is tricky and the testing prior to the "go live" deployment is critical to mitigating this issue. (And, of course, copious documentation should be captured for the resulting system.)

Application-specific Challenges

There are a couple of general points about the application portfolio that should be noted. As mentioned above, the lack of documentation on these aging systems makes the migration effort more difficult. The project team that drives a migration project must then resort to "mining" the actual application source code to determine exactly the behavior of the application.

Another important application-specific issue for consideration is discovering the integration requirements and dependencies of the application with other systems and databases. These integrations and dependencies must be clearly identified and, if still needed, they must be re-connected (possibly rebuilt) and made operational along with the migrated system.

Running Parallel Systems

For a short while, there may need to be some parallel processing between the mainframe application, while it is still being used in production, and the newly migrated system, on the new platform. Planning and executing this parallel processing will be a challenge, and will require extra time and attention to make it successful.

Another example of when you may choose to run parallel systems is if you want to achieve quick reductions in mainframe processing consumed by moving the development and test environments to an Azure-based emulator while keeping the production system on the mainframe for the interim.

Data Integrity

Moving the contents of large databases is very challenging on a number of levels. Typically, a database "cleanup" will be necessary to ensure that the contents of the new target database is as accurate, and as complete, as possible.

A mainframe modernization project is a good time to transform, correct and validate the organization's data.

Speed to Completion

In almost every project, speed will be a top priority. The costs and complexities of extended project cycles can have an enormous negative impact in tangible and intangible terms. As project cycles get extended, staff attrition can become a big issue and staff fatigue may also become a factor.

Paying for a continuation of the primary production system and funding the development efforts of the new system at the same time will have a temporary financial impact for as long as that duality continues. Getting to a "go live" status quickly and efficiently with the new system, and retiring the old system, will keep unexpected costs to a minimum.

Project Funding

It is very important for any modernization project to be properly funded and supported by the business management team and the executives. This support is essential to maintain project continuity and funding throughout the project cycle. Since we stated earlier that speed will be a factor in the project execution, funding must be in place to sustain that speed.

Expertise

Mainframe migration projects come in many forms. In every case, a variety of specialist skills will be needed on the project team. These specialists may include business analysts working to mine and understand the business rules embedded in the legacy applications.

It will also include experts in specific programming languages, databases, networks, terminal devices and many other components of the total application portfolio that will need to be addressed over the course of the migration to the target platform. Staff must also be available to address any specific functionality or use case of the mainframe application environment.

All this technology must be transferred to the equivalent functionality on the target cloud platform and work as it did in the original mainframe environment. Thorough testing by the project team, followed by testing amongst the business users of the original mainframe application system, is an absolute requirement. Once testing is completed, a final performance and tuning (P&T) exercise will ensure that the new cloud deployment is performing at optimal levels.

Why Azure

Organizations keep discovering new and improved benefits for moving their mainframe (as well as other) workloads to the public cloud such as flexibility, scalability, automatic backups, automatic software upgrades, cost model optimization, versioning control and adding multiple security layers, just to name a few.

Azure is a recognized leader in the public cloud space with a diverse customer base and a broad range of use cases. They also have one of the largest compute capacities in use by paying customers. This attracts Microsoft-based and open source application developers as well as service providers to make their applications compatible or add their services to Azure.

The benefits of migrating your mainframe to Azure are:

Easy to Use – Azure is designed with simplicity in mind. You can request new services and host your applications using their simple to use web-based Microsoft Azure Portal. All their services are well-documented and there is a wealth of forums, white papers, and discussion boards.

Flexible - You can select from a wide variety of virtual environments where you choose the software and services your application requires. If you find that the environment selections are not adequate, you can simply provision different types of instances or add compute and/or storage on demand.

Cost-Effective - Azure services are billed using a consumption model; you only pay for the compute and storage resources you use with no upfront commitments and contracts. Alternatively, if you know you have a minimal level of needs, you can sign longer-term contracts for additional savings.

Reliable - With Azure, you are taking advantage of its highly redundant, worldwide computing infrastructure that is built to guarantee high availability that rivals or exceeds what you've come to expect from mainframes.

Scalable - Azure includes tools such as auto-scaling and load balancing that allow your applications to scale up or down, if you design/architect this into your solution. Azure's massive compute and storage infrastructure guarantees that resources will be available when they are needed.

Hi Performance - Azure offers a wide selection of compute and storage options to replicate or exceed the performance needs of your formerly mainframe-based applications. Compute and storage can be provisioned as they are needed, so if your application is CPU intensive you can have a larger CPU/IO ratio and vice versa.

Secure - Azure's infrastructure is designed as a secure foundation that can host millions of customers simultaneously, giving you control and customization via a wide array of configurable security options. Azure prevents unauthorized and

unintentional transfer of information between deployments in a multi-tenant architecture, using virtual local area network (VLAN) isolation, access control lists (ACLs), load balancers, and IP filters, along with traffic flow policies; network address translation (NAT) separates internal network traffic from external traffic.

Azure is deployed in Microsoft regional datacenters, which are protected by layers of defense-in-depth security that include perimeter fencing, video cameras, security personnel, secure entrances, and real-time communications networks, continuing through every area of the facility to each physical server unit.

Achieving the Positive Impact of Change

In any mainframe migration project, the results of a cloud-based application set may be daunting.

The change will impact the technical staff, as they will likely need to learn new skills.

The end-user community may not notice too many changes using a new system if the interfaces are preserved. In fact, the move to the cloud could fuel new innovation resulting in new capabilities down the road, which are likely not available to mainframe users today.

The overall impact of a successful mainframe migration project is a positive one for the entire organization. A new and better application portfolio, a cloud platform to enable innovation, and a large cost savings in the operational and systems software maintenance categories will be realized. It's not unusual to repurpose IT staff after redeploying the mainframe portfolio to the cloud. The cloud platform has many other benefits, but flexibility and cost takeout are at the top of the list.

Understanding Typical Unisys Architecture

Since their development in the late 1940s, general-purpose mainframes were the computing workhorses for more than 50 years. Over that time, each mainframe manufacturer continuously enhanced their unique architectures to outperform competitors and meet evolving business demands. IBM and Unisys eventually dominated the market and became the gold standards of mainframe computing. This Unisys to Azure Reference Architecture is part of the Astadia Mainframe-to-Cloud Series of architectures, whitepapers and webinars.

Unisys Mainframe Heritage

Unisys can trace its roots all the way back to 1886 and the American Arithmometer Company, which later became the Burroughs Corporation.

The Unisys Corporation of today was formed in 1986 when Burroughs combined with the Sperry Corporation, which was originally founded in 1910 as the Sperry Gyroscope Company.

Unisys is credited with developing the first general-purpose computers, known as BINAC and UNIVAC.

When Burroughs and Sperry merged to form Unisys, each company had its own line of mainframe computers, each with a loyal customer base. There were attempts to unify the two architectures and their respective technologies, but the two distinct systems survive to this day.

Unisys **ClearPath Libra** mainframes originated from the Burroughs line, while **ClearPath Dorado**'s heritage is Sperry. Even though there are distinct differences between these two sets of mainframe technologies, they are in many ways very similar and share many of the same basic characteristics.

Unisys Mainframe - ClearPath Libra (Burroughs)

Unisys Mainframe - ClearPath Dorado (Sperry)

Unisys

User Interfaces

Users access the mainframe application through a variety of means. They could use green screen terminal emulators that provide character mode interface (T27, UTS).

Alternatively, a variety of custom user interfaces could be built on top of the character mode interface that allows a more user-friendly interface to mainframe applications. One of such user interfaces could be a web-based or mobile application serving as a front end to the mainframe.

Batch

Mainframes provide batch environments that handle bulk data processing workloads. Jobs are submitted to the system (WFL, EFL) and processed with minimal operator interaction. Output from the batch jobs is spooled, printed and distributed to users.

Transaction Processing

Transaction processing is at the core of most mission-critical applications *with thousands or millions of transactions being processed daily*. Mainframes provide the online processing environment (COMS, TIP) that make this possible. Security, transaction integrity, predictable response times are of particular importance for this type of workload.

Programming Languages

Mainframes provide an assortment of programming languages to suit customer needs. Most applications are written in COBOL but other languages are also used: Algol, MASM, BIS/MAPPER, FORTRAN, Pascal, PL/I, etc. 4GL development products like XGEN and ABSuite (LINC) are also used to develop Unisys applications.

Data Files

Mainframes store data in files with different record organizations and media types. Data files can be sequential, direct access, fixed and variable lengths, blocked or unblocked, etc. Data files can be stored on disks, magnetic tapes, CDROMs, etc. For the most part data in these files are stored in EBCDIC (Extended Binary Coded Decimal Interchange Code), an eight-bit character encoding system used primarily on mainframes.

Databases

Mainframes provide high performance database management systems to support online mission critical applications. These databases can be hierarchical (DMSII), networked (DMS), or relational (RDMS), and they provide high levels of availability, integrity, consistency, reliability, security, and auditing. Database software makes intensive use of the computing and input/output capabilities of the mainframe to provide optimal response times.

Environmental software

Mainframes also run software to support the management, operation, application development, and security of the system. Software provided by Unisys and third parties cooperate to provide a robust environment for customer applications.

Scheduling software is used to manage the execution of work flow streams. Output management systems handle the collection, storage and distribution of reports to the users that require them. Source management systems are used to maintain application source code by tracking version as well as release lifecycles.

Security is tightly controlled at all levels of the mainframe software. Security software is designed to minimize the risk of data exposure and provide regulatory compliance.

# Unisys to Azure Reference Architecture

As a Reference Architecture, the following diagrams and discussion addresses a typical use case. However, each implementation is sure to have its own customizations and variations, which is why a thorough application portfolio inventory, assessment and rationalization is critical to a successful outcome.

Below, you will see a design that includes details such as Azure components, batch requirements, programming language conversions and replacements, integration with external systems, 3rd-party software requirements, and planning for future needs.

In an actual project, you would also consider any unique features that would necessitate custom-made solutions. We would recommend proof-of-concept conversions on application subsets to test the model selected, discover any weaknesses, and prove the viability of the design.
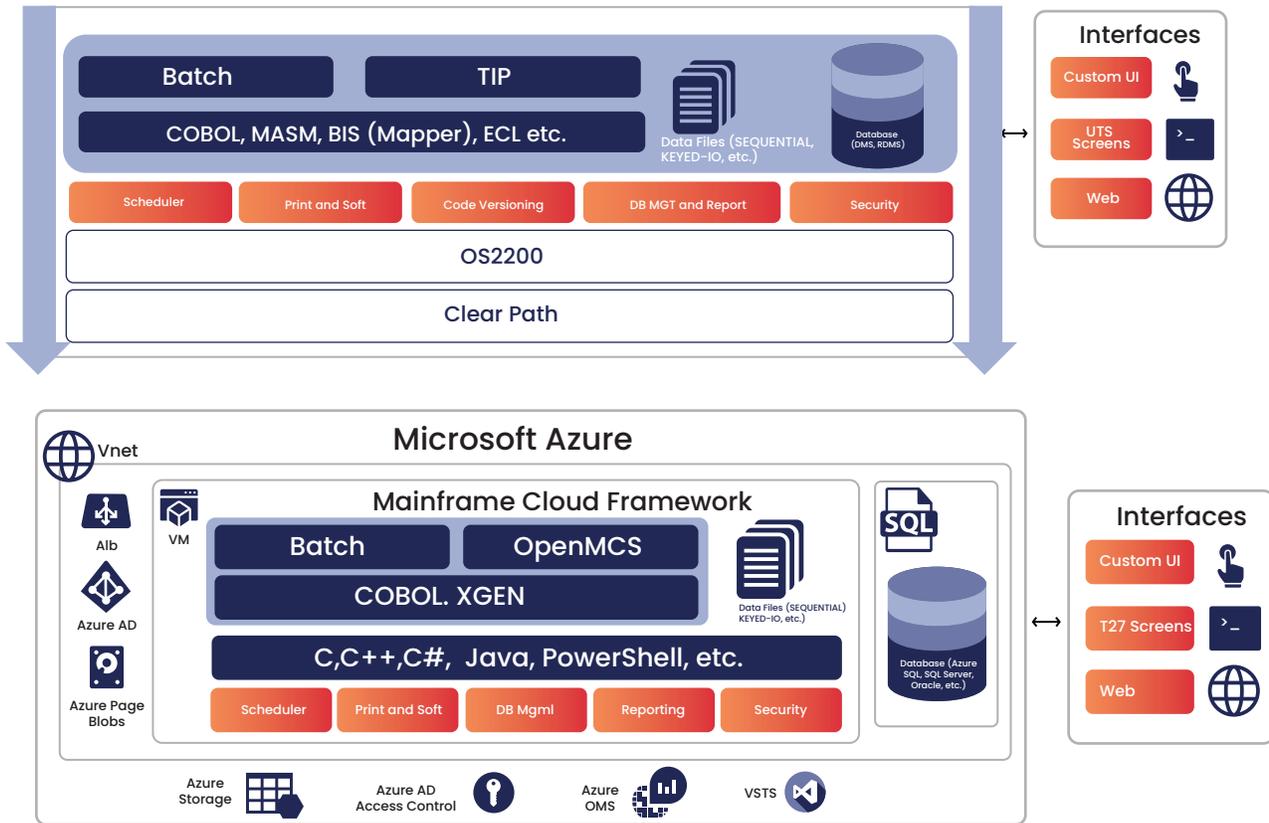
Migrating Unisys Libra (Burroughs) mainframe applications to Azure:

Unisys Mainframe - Clear Path Libra (Burroughs)

Microsoft Azure
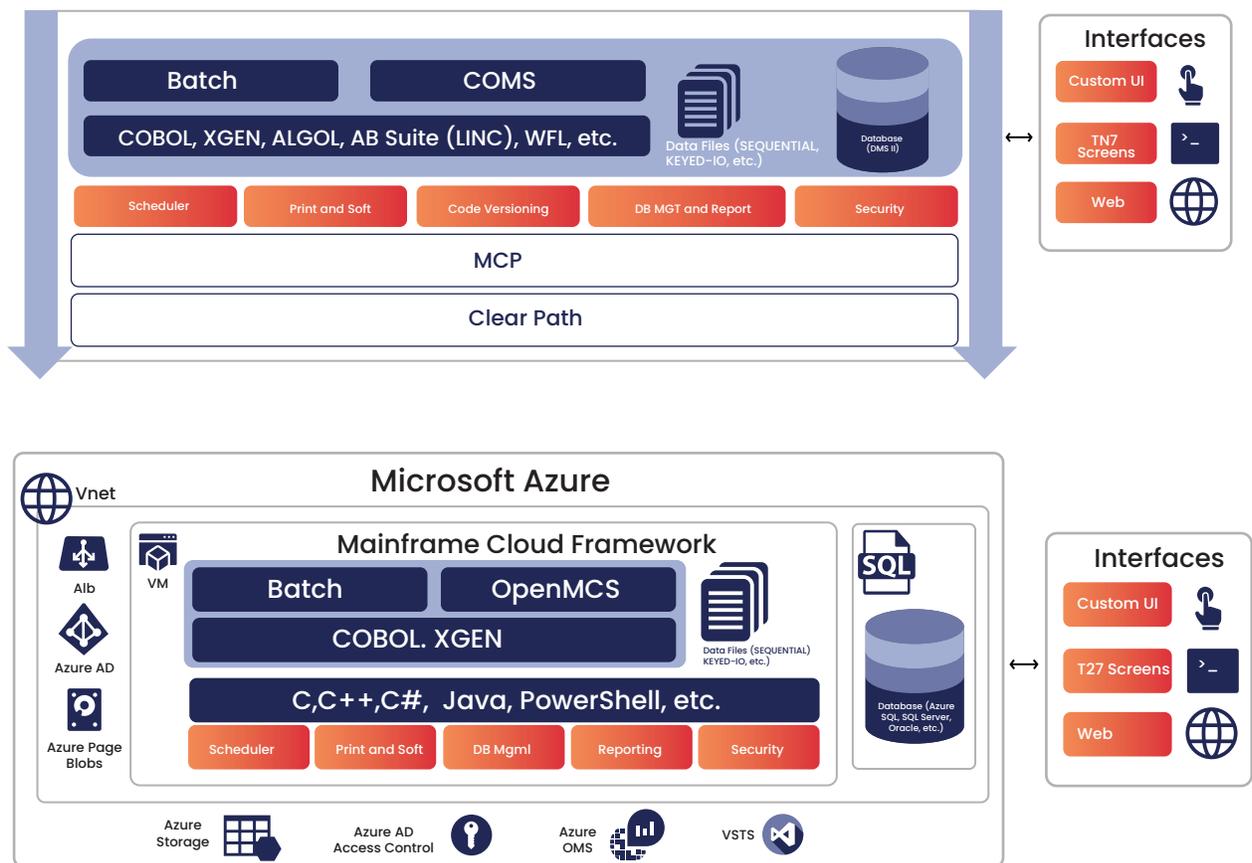
Migrating Unisys Dorado (Sperry) mainframe applications to Azure:

Unisys Mainframe - ClearPath Dorado (Sperry)

Code Modification

As part of any mainframe modernization project, there will be a need for partial or, in some cases, extensive code modification. Leveraging our past experience in developing Unisys productivity software (XGEN and OpenMCS), Astadia has developed its own code transformation and mainframe emulation technologies. We use these in concert with trusted third-party products to modify source code for deployment to Azure. What follows is a description of our approach.

Astadia employs an iterative, hybrid process of automated code conversion and human intervention. The technology behind the automation is Astadia's Rules-Based Transformation Engine.

This tool preserves the business logic and rules of legacy applications while removing proprietary code that can only execute in the source

environment and not in Azure. Its code migration filters ensure the preservation of mission-critical applications and back-end components such as trancodes, security policies, and message routing.

Though the Rules-Based Transformation Engine is a proven technology, Astadia augments our technology with years of hands-on migration experience and collaboration with partners. The combination of automation and human intervention ensures that legacy applications will work in Azure without sacrificing their original functionality.

Although every mainframe migration is unique, there are general source-to-target mappings for application components that apply to most projects, as shown in the following table.

Code Modification Mapping Table

Source	Target
MCP/OS2200 (Burroughs/Sperry)	Azure (Windows or Linux)
COMS/TIP	Astadia OpenMCS
COBOL, XGEN	COBOL, XGEN, Java, C#
Algol, MASM	COBOL, C#, Java, or mapped to OS functions
AB Suite (LINC)	C#,Java
BIS (Mapper)	C#,Java
WFL, ECL	VBScript, PowerShell, other scripting

These mappings are only a guideline for the most common mainframe technologies. Other technologies are addressed on an as-needed basis.

Database Migration

In parallel with code modification, data specialists will need to perform a thorough analysis of the legacy databases and files, and develop a detailed data migration strategy.

We recommend an iterative extract, transform and load process to identify potential data-typing issues, develop fixes, and collaborate with the application subject matter experts to validate their efficacy. This iterative process continues until every issue is eliminated.

In most cases, hierarchical and flat file data structures will be replaced with RDBMS solutions, but other solutions may be implemented to address unique technical requirements or preferences.

After the target database and file structures have been created and validated, static data can be migrated to the Azure production environment. For dynamic or other data that is created and/or modified frequently, a data migration strategy must be implemented as part of production cutover process.

Like the application component mapping above, there are general source-to-target data mappings employed by most mainframe-to-Azure migrations:

Database Migration Mapping Table

Source	Target
DMS	Azure SQL, SQL Server, Oracle, DB2 LUW, MySQL etc.
DMSII	Azure SQL, SQL Server, Oracle, DB2 LUW, MySQL etc.
RDMS	Azure SQL, SQL Server, Oracle, DB2 LUW, MySQL etc.
Flat files	Flat file or Azure SQL, SQL Server, Oracle, DB2 LUW, MySQL etc

These mappings are only a guideline for the most common mainframe database and file technologies. Other technologies are addressed on an as-needed basis.

Azure/Unisys Integration and Parallel Operations

For some organizations, a one-time deployment and cutover of all mainframe applications to Azure simply isn't feasible.

There may also be organizations who prefer to move applications one at a time or in smaller groups as a way of gradually embracing Azure as a solution for their big-iron applications.

Still others may intend to keep their mainframes indefinitely for a subset of strategic, mission critical applications while migrating less critical applications to Azure as a means of reducing their costs or delaying upgrades to existing mainframe hardware.

There's good news here - it is possible to do a phased rollout of migrated applications and still have ongoing communication and integration with applications residing on the Unisys mainframe. This kind of mixed environment can be achieved with the proper planning, and will generally resemble the image below.

Unisys mainframe cross-premises integration with applications & data already migrated to Azure
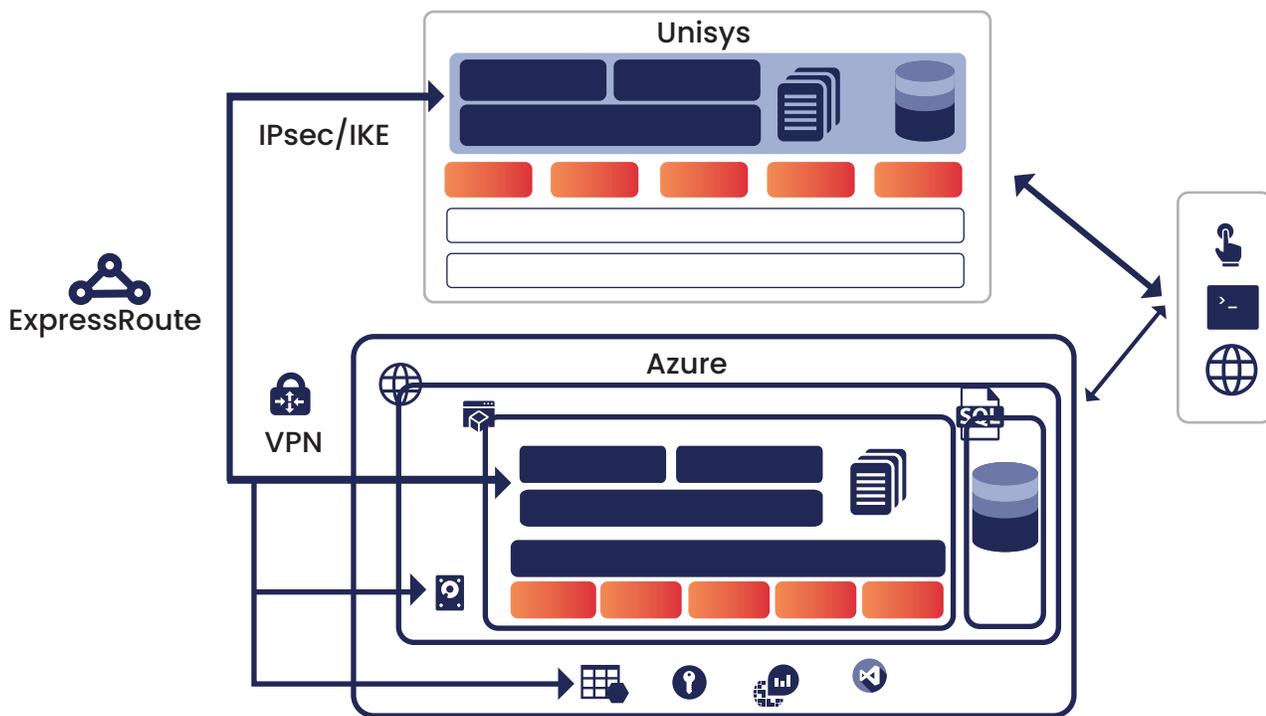
Understanding Microsoft Azure

Microsoft Azure has provided the technology and services that make running mainframe applications in the cloud a safe, secure, reliable way to achieve high performance results and fuel future innovation for the organization.

There are specific elements of Azure that are relevant to a mainframe modernization project. Below, we address some of these – this is not the extent of all Azure services nor is it meant to exclude the use of other Azure service offerings.

Let's begin with the following description from the Microsoft Azure website:

Azure is built on the same technologies that you (and millions of others) already trust: Windows Server, SQL Server, Active Directory, Hyper-V, and more. This means a consistent, familiar, and stable environment. Move your virtual machines between on-premises and the cloud as often as your business needs dictate — no conversion needed.

Help protect your business-critical applications with managed cloud backups and Disaster Recovery as a Service built in. For applications spanning on-premises and Azure, take advantage of ExpressRoute for a private, fast, and reliable connection.

With unparalleled compute and storage performance, Azure is where to run your demanding workloads. G Series virtual machines and >80,000 IOPs of Premium Storage performance deliver a powerful combination for your data-intensive workloads, including SQL Server, SQL Data Warehouse, Oracle databases, NoSQL databases, and SAP SD Standard Application. In fact, we broke a world record for running an SAP SD Standard Application in a public cloud environment. Azure G and D Series virtual machines run on the latest Intel cloud technology, which are designed with the latest Intel Xeon processors. For you, that means record-breaking performance and reliability that can handle your data and compute-intensive applications.

You don't need to be an expert on configuring, deploying, and maintaining your business applications—because we're here to help. Choose from a growing number of pre-packaged templates to get started immediately. These templates are based on tried and tested configurations, and help you

easily set up complex environments for applications, including SharePoint and Dynamics AX.

We're enabling a hybrid world because that's what you want. You've asked for the ability to move existing Windows Server and Windows licenses to Azure, just as you can with other popular Microsoft products, such as SQL Server and SharePoint. And we're pleased to say that we're making that possible.

Azure is not just about Microsoft software. It is an open and flexible cloud that supports multiple operating systems, programming languages, frameworks, tools, databases, and devices—it's about your business needs. We continuously work with industry-leading software vendors to certify their applications on Azure and to give you peace of mind. We have unique relationships with SAP, IBM, Oracle, and Red Hat so that you get a first-class support experience. In addition, Azure Marketplace provides more than 3,400 certified applications that are ready to be deployed.

We are committed to giving you the best pricing and flexible options. For many apps, you can bring your existing licenses to Azure or enjoy a pay-as-you go model. You can also take advantage of various pricing plans that fit your unique needs. If you have predictable workloads with well-defined compute requirements, you can take advantage of the Compute Pre-Purchase Plan—and save up to 63 percent on your compute costs. And if you're moving a significant number of workloads to Azure, but at a moderate or less predictable pace, the Compute Option provides great pricing with more budget flexibility. The more you use Azure compute, the more you save.

In the following section, we'll take a deeper look at the Azure portion of the Reference Architecture and the components identified.

Azure Architecture Supporting the Migration of Unisys Applications
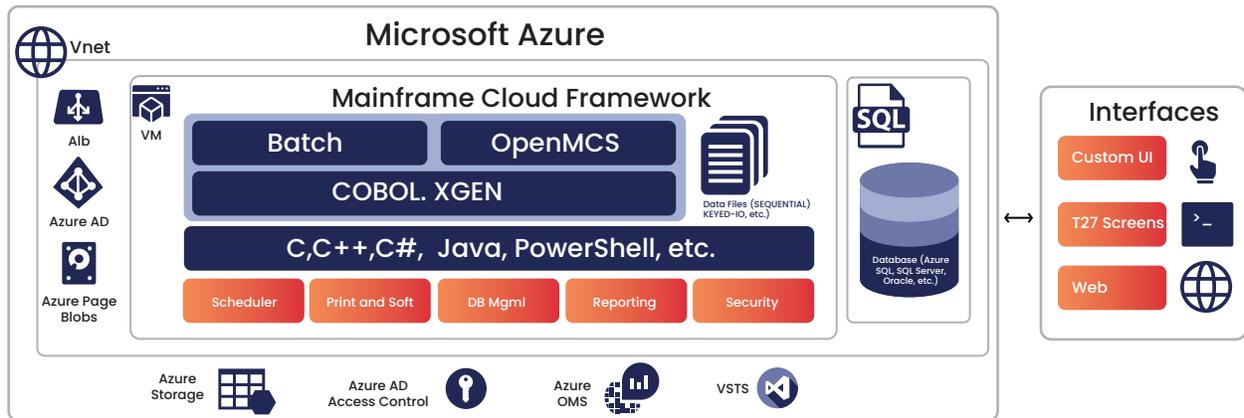
Your Cloud Environment

Azure Virtual Network (VNet) lets you provision a logically isolated section of Azure where you launch and manage interconnected resources in a virtual network that you define. It's your private area within Azure.

You can think of this as the fence around all the systems you have in Azure. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways. You can also integrate your on-premises network using a VPN tunnel or Azure ExpressRoute Service.

Computing Resources

Azure Virtual Machines (VM) is one of several types of on-demand, scalable computing resources that Azure offers. Typically, you would choose a VM when you need more control over the computing environment than the other pre-configured choices offered. For mainframe workloads, this serves as the foundation upon which your applications sit. It's the container that holds the operating systems, mainframe emulators, application executables, and other supporting software that make up your application.

Depending on your specific circumstances, you may separate some pieces into their own Azure VM instances or you may run everything in one VM. For instance, maybe you'll have an Azure VM dedicated to batch COBOL and another dedicated to Online. You may even segregate VMs by application, application types or maybe by the databases they access. Each organization has unique needs and specific requirements. The

number of VMs that your applications use can scale up and out to meet your needs.

You'll also configure VMs dedicated to specific tasks, such as development and maintenance of your migrated applications and various testing environments.

Storage

Azure Storage services can be thought of as hard drives for storing your data. It provides the storage foundation for Azure Virtual Machines, is massively scalable and can store and process hundreds of terabytes of data to meet the demands of your migrated applications.

Azure Storage uses an auto-partitioning system that automatically load-balances your data based on traffic. This means that as the demands on your application grow, Azure Storage automatically allocates the appropriate resources to meet them

Azure Storage is accessible from anywhere in the world, from any type of application, whether it's running in the cloud, on the desktop, on an on-premises server, or on a mobile or tablet device. Services include Disk storage, Blob storage, Queue storage, and File storage – each with its own specialized characteristics.

Databases

Legacy relational data will reside in database VM instances. This includes any hierarchical and flat file data that's been converted to relational. For example, all your DMS and DMSII data would be converted to relational data and stored in a VM running SQL Server or other database products like Oracle or MySQL. RDMS data would also be migrated to one of these VMs.

Azure SQL is optimized for database performance. It's cost-efficient, has resizable capacity, and is designed to reduce time-consuming database admin tasks. However, it's best suited for new databases rather than legacy data. You'll most likely utilize dedicated VMs running SQL Server or another RDMS product.

Azure also offers other database services like SQL Data Warehouse for data warehousing, SQL Server Stretch Database to stretch on-premises databases to the cloud, and Azure Cosmos DB for globally distributed, multi-modal databases.

An analysis of your existing legacy databases and applications will reveal all the changes required to migrate your data to Azure as well as which database services make the most sense for your migrated applications.

Load Balancing

Applications with a high volume of transactions require something to balance the workload. Azure Load Balancer (ALB) does just that. It automatically distributes incoming application traffic across multiple VM instances to achieve scalability, high-performance, and fault tolerance in your migrated applications. It provides the load balancing capability needed to route traffic evenly among your applications and keep them performing efficiently.

Security

Microsoft's Active Directory (AD) is a Microsoft brand for identity related capabilities. In the on-premises world, Windows Server Active Directory (WSAD or simply AD) provides a set of identity capabilities and services. Azure Active Directory (Azure AD) is Microsoft's vehicle for providing identity management in the cloud. While there are other possibilities, like LDAP, Azure AD is most likely where you'll map your legacy application user IDs, passwords, permissions, etc.

Azure AD also includes a full suite of identity management capabilities including multi-factor authentication, device registration, self-service password management, self-service group management, privileged account management, role based access control, application usage monitoring, rich auditing and security monitoring and alerting. These capabilities secure your migrated applications, streamline IT processes, cut costs and help ensure that corporate compliance goals are met.

Monitoring and Management

Every IT system needs to be monitored and managed. Azure Operations Management Suite (OMS) is a collection of cloud-based services allowing you to monitor and manage the resources now running the legacy applications you've deployed to Azure.

You use this tool to collect and track metrics, monitor log files, set alarms, backup and restore critical data, and automatically react to changes in your Azure resources. These services are used to resolve problems quickly and keep your migrated applications running smoothly – much like you do on the mainframe today. OMS also manages on-premises resources, so you'll have a unified environment to manage all your IT operations no matter where they physically reside.

Source Control

Just as you have products and processes to control your application sources and manage application releases on your mainframe today, you need to have a similar set of tools in Azure.

Visual Studio Team Services (VSTS) is a fully-managed source control service providing secure and private Team Foundation and Git repositories. It eliminates the need to operate your own source control system or worry about scaling its infrastructure.

VSTS is where you'll store your migrated application source code and binaries, new source and binaries, an anything else you want to archive.

Ensuring Project Success

Astadia's Legacy Modernization practice has more than 25 years of experience migrating legacy applications to modern platforms. Since mainframe applications are the mission-critical systems of the enterprise, Astadia goes to great lengths to ensure a thorough and complete project plan is developed for each legacy modernization project we undertake.

Astadia's methodology recognizes the organizational impact that any project of this nature will have on day-to-day operations, as well as the financial and business implications for organizations in both the short and long term. Return on Investment (ROI), and Total Cost of Ownership (TCO), are carefully calculated during this process, and are closely managed throughout the project lifecycle.

Astadia's Mainframe-to-Azure Success Methodology has been refined over the course of 200+ successful legacy migration projects, and has become an industry leading approach for our medium and large-scale mainframe clients.

Astadia Mainframe-to-Azure Success Methodology
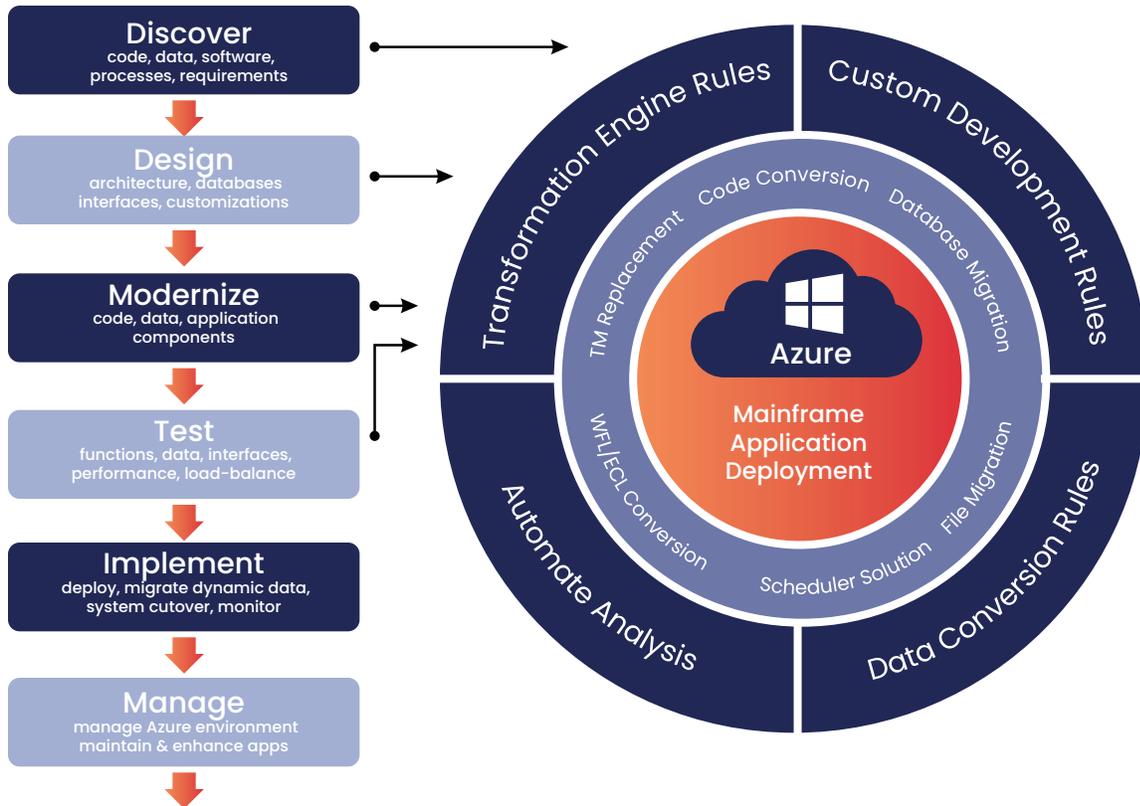
Discover

Catalog and analyze all applications, databases, networks, platforms, and processes in the client's portfolio. Document the interrelationships between applications, and all external integration points in the client's configuration. This is a key input to Application Portfolio Management and Application Rationalization.

Design

Astadia's project team analyzes source code, data structures, end-state requirements, and Azure cloud components to design and architect the solution. The design includes details such as types and instances of Azure components, transaction loads, batch requirements, programming language conversions and replacements, integration with external systems, 3rd-party

software requirements, and planning for future requirements.

Modernize

Astadia employs an iterative, hybrid process of automated code conversion and human intervention to perform the necessary application changes. The technology behind the automation is Astadia's Rules-Based Transformation Engine. This tool preserves the business logic and rules of the client's legacy applications while removing proprietary code that can only execute in the source environment and not in Azure. While a least-change, least-risk approach is employed, some source code or supporting components may be converted to new languages for technical reasons or to comply with client preferences.

Test

Since most mainframe-to-Azure projects are typically an as-is migration of applications, testing can focus primarily on two areas: the components that have been changed or replaced, and application performance. For the most part, "parallel" testing is sufficient; that is, the results of operations in the source environment must produce the same values in the target environment, except where differences are expected due to platform changes. For performance testing, the focus is on ensuring the user experience is as good or better than the legacy environment and batch processes complete within acceptable timeframes.

Implement

When migrated applications have been tested, verified, and optimized, the process of deploying those applications may begin. In reality, many deployment activities are initiated in parallel with earlier phases – things like creating and configuring Azure component instances, installing and configuring mainframe emulation software, migrating static data, and other infrastructure or framework activities. In some cases, environments may be replicated to achieve this, or existing environments may be re-purposed. The specifics of this may depend upon application and data characteristics and client preferences. After dynamic data is migrated and validated, cutover to Production mode can be completed.

Manage

Astadia offers a full range of managed services solutions. Having gained significant application knowledge from the architecting and implementing the Azure solution, Astadia is well suited to take on the burden of managing and maintaining the migrated applications and their Azure environments or the dual environment in the case of a partial migration. This offers clients an opportunity to focus their development efforts on strategic initiatives as well as address concerns of finding programmers skilled in maintaining the legacy components still in use.

Conclusion

Microsoft Azure and Astadia combine to create a perfect next generation platform for your mainframe applications portfolio.

Once the mainframe application set has been fully deployed on Azure, you will have the freedom to re-engineer traditional applications in to a more contemporary computing style, modernize legacy interfaces and integrate with other applications. In addition, many new services, like mobile and wireless, can be easily connected to the cloud platform, thus enhancing the overall power of your new cloud computing environment. Your investment will serve to support all the needs of your enterprise and the future requirements of your business.

You don't have to tackle this alone. Astadia has the experience, skilled experts and the technology to successfully help you complete Legacy Migration projects of all scopes and sizes.

Astadia would be happy to hear from you about your specific Legacy Migration needs and how we may be of service to you as you prepare to leverage Azure.

To connect with an Astadia expert, please contact us at:
Email: info@astadia.com
Web: www.astadia.com
Phone: (877) 727-8234