

Test Study: Implementing a Gas Utility's Network Management System on SAP HANA in AWS

Last generated: April 05, 2024



Table of Contents

- **Test Study**
 - Introduction and overview 2
 - Tested workflows 4
 - Physical architecture 11
 - Design choices and considerations 16
 - Test methods and results 21
 - Using this test study 28

Network Management System: Gas Utility (SAP HANA)

This test study was conducted in August 2023 and evaluates the [reference architecture](#) for a Network Management System with:

- [Workflows](#) representative of a gas utility
- An enterprise geodatabase configured with SAP HANA
- Amazon Web Services (AWS) cloud infrastructure.

The system presented in this test study was designed at a physical level, and incorporates several [design choices](#) that are described in detail. The system was deployed and tested for both performance and end-user experience, with a target design load of 15 ArcGIS Pro Editors and 200 ArcGIS Web Users (General User personas)

Note:

It is important that you become familiar with the [reference architecture](#) *before* the test study, as there is important information and resources that are not duplicated here.

Overview of capabilities

A foundational network management system delivers the following capabilities as defined in the [conceptual architecture](#):

- Service-based editing of relational data, including geometry and attributes.
- Data collection workflows using forms and/or maps.
- Complex utility network management using the ArcGIS Utility Network as an advanced data model.
- Viewing and querying network assets through web applications.
- Performing upstream & downstream trace analytics.

Software list

The system capabilities are delivered through the following software, deployed, and tested as part of this test study, at the listed versions with all available patches applied:

Data characteristics

- [ArcGIS Pro 2.9.4](#) (latest version [here](#))
- [ArcGIS Enterprise 10.9.1](#) (latest version [here](#))
- [ArcGIS Monitor 2023](#) (latest version [here](#))
- [ArcGIS License Manager 2022](#) (latest version [here](#))
- [ArcGIS Online](#)
- SAP HANA 2.9 SPS 06 rev63

Note:

The software versions listed above were the latest available when this system test study was deployed and tested. Esri strongly recommends using the latest available software versions when possible.

The design was deployed on a Microsoft Windows operating system, apart from the database server, which was deployed on SUSE Linux.

Data characteristics

This test study tested a gas utility dataset with a geographic extent of roughly 55,000 square miles, and with an overall size of approximately 45 GB. The dataset contained roughly nine million features, 700 sub-networks, and 300,000 gas meters.

Tested workflows

To ensure that workflows are useful for system design purposes, they need to represent real user experiences, and the actual steps that users will take in interacting with the system. These workflows represent some of the foundational activities required to create, access, and maintain a gas network. The contents of the workflows were defined by working with experienced staff and Esri customer feedback to identify the specific steps, sequencing and type of activities involved in each workflow.

Developing a workflow as a detailed, stand-alone set of steps allows them to be repeated, iterated on to tweak operations, or adjusted to improve performance or user experience. Once each workflow is clearly defined, it is used consistently throughout the remainder of testing to establish the performance baseline and conduct load tests. If your organization does not have clearly documented, standard workflows with individual steps defined, consider using these workflows as a starting point that can be adjusted based on your needs.

This implementation of the Network Management System's reference architecture was load tested for the following ten (10) gas utility workflows:

- [Create a service](#)
- [Remove a service](#)
- [Replace a main](#)
- [Extend a main](#)
- [Trace analytics](#)
- [Modify a feature](#)
- [Linear referencing](#)
- [Query assets](#)
- [View network](#)
- [Summarize assets](#)

These workflows support both editor and general user [user personas](#).

Editor workflows

Editors perform edit operations on the data, either editing attributes and geometry on single records or editing attributes in bulk. Editors may range in expertise and overall comfort with technology and may also perform edits in the office or in the field.

Create a service

This workflow represents the set of tasks an editor user persona performs to extend the grid to provide gas to a new customer.

1. Sign in to ArcGIS Pro and open project
2. Locate asset
3. Zoom to service Area of Interest (AOI)
4. Create and change version
5. Create metal service tap
6. Create company service pipe
7. Create plastic curb valve
8. Create customer service pipe
9. Create meter group
10. Zoom to distribution pipe extent
11. Validate topology
12. Reconcile
13. Post
14. Change version to default
15. Validate topology in default

Remove a service

This workflow represents the set of tasks an editor user persona performs when a service needs to be removed from the network for any reason, such as a demolition.

1. Sign in to ArcGIS Pro and open project

2. Locate asset
3. Zoom to customer pipe
4. Create and change version
5. Abandon customer pipe
6. Edit vertices of customer pipe
7. Abandon riser
8. Abandon meter
9. Abandon plastic curb valve
10. Modify riser and meter
11. Zoom to company pipe
12. Abandon company pipe
13. Edit vertices of company pipe
14. Zoom to dirty area
15. Validate topology
16. Reconcile
17. Post
18. Change version to default
19. Validate topology in default

Replace a main

This workflow represents the set of tasks an editor user persona performs when a gas utility needs to replace a gas main due to its age, condition, or other reasons.

1. Sign in to ArcGIS Pro and open project
2. Locate asset
3. Zoom to pipe
4. Zoom to AOI
5. Create and change version
6. Outrage trace

7. Split the pipe
8. Abandon the split pipe
9. Edit vertices of split pipe
10. Add two controllable valves
11. Add coated steel distribution pipe
12. Modify terminal connections – pipe 1
13. Modify terminal connections – pipe 2
14. Modify terminal connections – pipe 3
15. Zoom to pipe
16. Validate topology
17. Reconcile
18. Post

Extend a main

This workflow represents the set of tasks an editor user persona performs when a gas utility needs to extend a main, such as due to new construction.

1. Sign in to ArcGIS Pro and open project
2. Locate distribution pipe
3. Zoom to AOI
4. Create and change version
5. Move metal end cap
6. Abandon metal end cap
7. Add transition
8. Zoom to new extent
9. Add plastic pe distribution pipe
10. Add plastic end cap
11. Validate topology
12. Reconcile

13. Post
14. Change version to default
15. Validate topology in default

Trace analytics

This workflow represents the set of tasks an editor or general user persona performs when they need to understand potentially impacted upstream or downstream areas by a main break, burst pipe, or other reasons.

1. Sign into ArcGIS Pro and open project
2. Locate asset
3. Zoom to selection
4. Run upstream trace
5. Run downstream trace

Modify a feature

This workflow represents the set of tasks an editor user persona performs when a service pipe has been moved from its currently documented location during construction, or to perform simple data correction.

1. Sign in to ArcGIS Pro and open project
2. Locate asset
3. Create and change version
4. Select features
5. Edit attributes
6. Move location of service pipe
7. Validate version
8. Reconcile
9. Post
10. Change version to default
11. Validate default

Linear referencing

Linear referencing is the method to store and geographically locate data using relative positions along a measured line feature without the need to explicitly use x,y coordinates or an address.

1. Sign in to ArcGIS Pro and open project
2. Select pipe by attributes
3. Zoom to selected pipe
4. Zoom to AOI
5. Run reverse route

General user workflows

General users are typically considered “viewers” within the system, and primarily discover and use content created by others within the organization.

Query assets

This workflow represents the set of tasks a general user persona would perform in a web application to identify and view assets of a certain type or by a set of attributes.

1. Sign in to ArcGIS Enterprise
2. Open web application
3. Open query builder
4. Enter query criteria
5. Run the query
6. Zoom to results

View network

This workflow represents the set of tasks a general user persona would perform in a web application to view and search for a specific device within the network.

1. Sign in to ArcGIS Enterprise
2. Open web application

General user workflows

3. Enter device ID in search box
4. Select result from suggestions
5. Zoom to device

Summarize assets

The steps below represent a use case where an ArcGIS Dashboard is open on a screen and refreshes every 30 seconds.

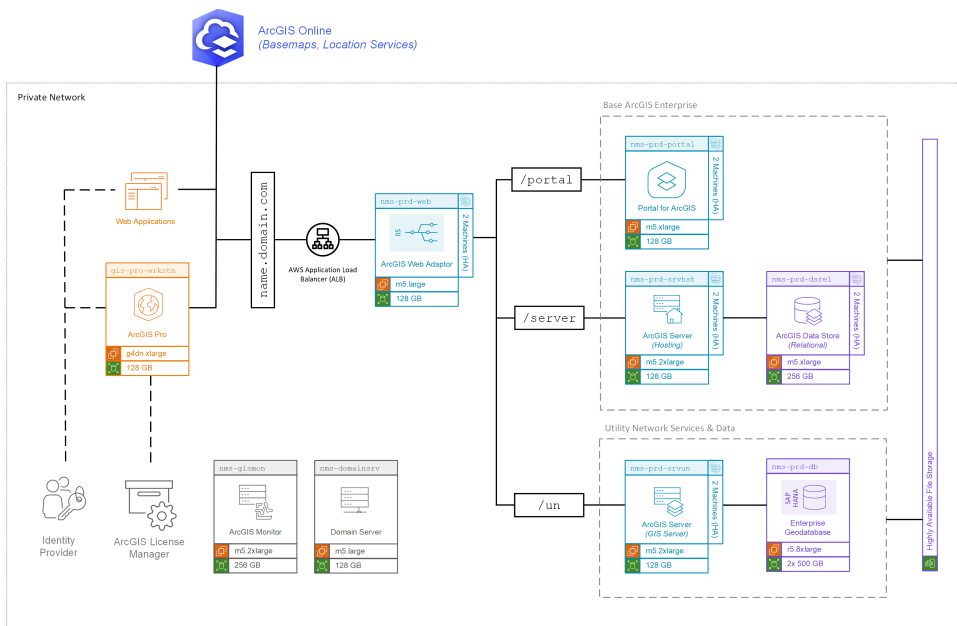
1. Sign in to ArcGIS Enterprise
2. Open dashboard
3. Automatic refresh (30 seconds)

Physical architecture

This architecture was designed with:

- A small/medium-sized gas utility in mind
- Support for [workflows](#) with a target design load of 15 ArcGIS Pro editors and 200 ArcGIS web users (general user personas).
- An enterprise geodatabase configured with Microsoft SQL Server
- AWS cloud infrastructure

The system was designed for and [tested](#) against the specified workflows, with adjustments to the machine types and sizes as needed based on test results.



Note:

For more information on the software components and key interactions in this architecture, please review the Network Management System [reference architecture](#).

Machine types & sizes

The following represent the machine sizes chosen and validated for the scope and purpose of this test study. However, it is highly recommended to follow a complete design process to account for your business and technical requirements.

Esri offers [system architecture design](#) services should you need help determining all of the different factors relating to your organization's physical design- such as networking, storage, system environments, and sizing. Minimum system requirements for each component are listed in the respective software [documentation available online](#).

Desktop (ArcGIS Pro & web apps)

- 3 machines (used in [testing](#))
- G4DN.xlarge instance type
- 2 CPU (4 vCPU)
- 16 GB RAM
- 125 GB Disk

Portal for ArcGIS

- 2 machines
- M5.xlarge instance type
- 2 CPU (4 vCPU)
- 16 GB RAM
- 128 GB Disk

ArcGIS GIS Server

- 2 machines
- M5.2xlarge instance type
- 4CPU (8 vCPU)
- 32 GB RAM
- 128 GB Disk

ArcGIS Server (hosting server)

- 2 machines
- M5.2xlarge instance type
- 4CPU (8 vCPU)
- 32 GB RAM
- 128 GB Disk

ArcGIS Data Store (relational)

- 2 machines
- M5.xlarge instance type
- 2 CPU (4 vCPU)
- 16 GB RAM
- 256 GB Disk

ArcGIS Web Adaptor

- 2 machines
- M5.large instance type
- 1 CPU (2 vCPU)
- 8 GB RAM
- 128 GB Disk

ArcGIS Monitor

- 1 machine
- M5.2xlarge instance type
- 4 CPU (8 vCPU)
- 32 GB RAM
- 256 GB Disk

File storage

- 1 instance
- 64 MB/s throughput
- 512 GB Disk

Database

- 1 machine
- R5.xlarge instance type
- 16 CPU (32 vCPU)
- 256 GB RAM
- 128 GB Disk

Domain sever

- 1 machine
- M5.large instance type
- 1 CPU (2vCPU)
- 8 GB RAM
- 128 GB Disk

Additional infrastructure considerations

The following are additional areas of consideration when designing a Network Management System and an explanation of some infrastructure choices made for this test study.

Application load balancer (ALB)

At least one third-party load balancer is required in a highly available ArcGIS Enterprise deployment—they handle client traffic to your portal and server sites and internal traffic between the software components. Although the ArcGIS Web Adaptor is considered a load balancer, it is insufficient on its own to serve as a load balancer in a [high availability](#) configuration. Therefore, in this test study, an Amazon Web Services (AWS) Application Load Balancer was used.

Shared storage

To successfully implement a highly available ArcGIS Enterprise deployment, the configuration stores must be stored in a highly available, shared location. This ensures this data remains accessible even if one server fails, providing uninterrupted service to end-users. Additionally, shared storage simplifies data management in a multi-machine deployment and improves scalability by centralizing data storage and allowing for expansion as needed.

System components not included in the diagram

Note that while antivirus software and AWS networking components are not illustrated in this diagram, they were present in the test study.

Design choices and considerations

The following considerations are organized around the [architecture pillars](#) of the [ArcGIS Well-Architected Framework](#). The appropriate application of best practices and architectural approaches in each of these technical areas contributes significantly to the successful design and implementation of well-architected systems.

Performance and scalability

Workload separation

The choice to design for [workload separation](#) was made to help achieve an optimal distribution of compute resources across the system. In the test study, editing requests generally took longer to process than standard map requests, so the choice to isolate editing workloads with dedicated compute resources in the form of a separate ArcGIS GIS Server site was made. Additionally, isolating the system components themselves onto different machines helps to ensure they don't compete for system resources and allows for the opportunity to tailor machine types and sizes to the system requirements of each component.

GPU-Enabled desktop machines

Selecting the proper GPU (Graphics Processing Unit) is essential for ensuring the performance of ArcGIS Pro in a virtualized environment. Tests revealed that adding a dedicated GPU to ArcGIS Pro virtual machines significantly improved end-user productivity and produced a net reduction in cost when operational expenses (labor costs) are considered. Learn more about GPU hardware selection and [ArcGIS Pro virtualization](#) in the ArcGIS Architecture Center.

Watching for vCPU: CPU in the cloud

It's important to understand the ratio of Virtual CPU (vCPU) to physical CPU when making design decisions so system components may be assigned appropriate resources. There is a 2:1 ratio of vCPU:CPU for all machines in the diagram, but some virtualization options may have different ratios, such as 1:1. These decisions may also have [Esri licensing](#) implications. Some examples of public cloud ratios include [AWS](#), [Azure](#), and [GCP](#).

GIS services configuration

Proper configuration of GIS services is critical to system performance and user experience satisfaction, and the misconfiguration of GIS service instances can introduce problems or reliability challenges in a system. For example, if the number of instances for a map or feature service are set too low, it can result in long client wait times and timeout errors.

Setting the instance count too high, however, can consume excessive machine resources, limiting the number of services that can be deployed on a fixed hardware configuration. When the maximum instance setting is higher than the minimum, the system can automatically add new instances in response to demand, but this can also be problematic because incoming requests must wait for the instance to start. For any system, it is important to understand service usage so that instance numbers and server resources can be adjusted to provide optimum performance.

In this test study, the ratio of service instances to physical CPU cores was set to 2:1 for each relevant service, with the minimum and maximum instances settings configured at that same value. The instance usage was monitored to determine when the system was overloaded. For example, at 8x design load, the service instances for a service on the hosting server were observed as active for 99% of the test period, which led to high wait times for read-only services. The services in this test were configured for dedicated instances. Learn more about [configuring service instance settings](#).

In this test study, the utility network services were configured as follows:

- Minimum number of instances per service: 8
- Maximum number of instances per service: 8

The total number of available instances was 16 because there were two ArcGIS GIS Servers in the site. The hosting servers were configured as follows:

- Minimum number of instances per service: 6
- Maximum number of instances per service: 6

The total number of available instances was 12 because there were two ArcGIS GIS Servers in the site.

The specified service timeouts were configured as follows:

- Maximum time a client can use a service : 600 seconds
- Maximum time a client will wait to get a service : 600 seconds

- Maximum time an idle instance can be kept running: 1800 seconds

Reliability

Backups

Backups are critical for Network Management Systems. Refer to the [reference architecture](#) for more information. While the tested design was not a production system, machine snapshots and database backups were captured for each test run and before making any changes to the system. Virtual machine snapshots were taken before and after any change in the environment (such as resizing a machine, installing a patch, or updating Windows). Snapshots were then cataloged to enable either:

- Roll back of a specific machine to a specific point in time
- Roll back of the entire environment to a specific point in time

High availability

The choice to design this system with a [high availability](#) configuration of ArcGIS Enterprise components was made based on business and technical system requirement, along with other organizational goals such as achieving uninterrupted operations and minimizing downtime. This configuration is illustrated in the design with redundant system components and a cloud-native, highly available file store for file storage. This test study did not configure a highly available database for testing purposes, though relational database vendors have a variety of methods to approach high availability including cloud-native services.

Note:

Keep in mind that high availability configurations can significantly increase infrastructure and operational costs of the system, and requires specialized skills to be successful. Learn more about [design choices and considerations](#) with regard to high availability for a Network Management System.

Observability

To perform successful system validation and deliver meaningful results, [system monitoring and telemetry capture](#) were key aspects to the test study.

ArcGIS Monitor and enterprise IT monitoring tools like Windows Performance Monitor were used to monitor the system's performance and capture telemetry on its behavior under certain conditions. Logs were collected across different system components, including:

- IIS web server
- ArcGIS software components
- Windows Events
- ArcGIS Pro

Machine-level metrics such as CPU usage, RAM consumption, disk activity, and network activity were captured across all machines in the environment. Review the [test results](#) for more information.

Additionally, screen recordings were captured of conducted workflows to observe and assess end-user experience and productivity.

Automation

Because the scope of the test study was primarily focused on load testing, most types of automation that would be recommended for a production system (like scripting administrative tasks) were not employed. However, in your environment, administrative scripts can have significant value to workflows and operations. Any automation scripting should be tested in a lower environment before deploying to production.

In this test study, the primary application of automation was for the purpose of simulating requests during load tests. Multiple workflows were run with virtual users at scale with the ability to apply to different load sizes, as illustrated in the [test results](#).

Python scripts were used to perform analysis on and identify patterns in service wait times, ArcSOC utilization, response times, and failed requests to inform needed system changes. Python, PowerShell and SQL scripts were also used to restore the database to an original state after completing a load test.

Security

While security was not the focus of the test study, it is critical to consider [security](#) requirements early in the design process for any production system. ArcGIS software has been designed to work effectively within secure networks, including those that are fully disconnected from the internet. The test study design does include the use of an identity provider to provide proper authentication and authorization.

Related resources:

- [ArcGIS authentication models and providers](#)

- [Authorization and access models](#)
- [Secure network design](#)

Integration

While integrations were not within the scope of the test study, a Network Management System often requires integration with other enterprise systems like a Enterprise Asset Management (EAM), Customer Relationship Management (CRM) and Advanced Distribution Management (ADMS) systems. In addition to [standard integration considerations](#) with ArcGIS, the ArcGIS Utility Network capability has additional requirements to consider. Depending on the integration requirements, different APIs and/or SDKs may be supported. See [Journey to the Utility Network: Integrations Overview](#) for more information.

Test methods and results

Testing was conducted to validate that the design would perform as expected and support the workflows, users, and intended load. System tests provide the opportunity to discover and correct problems during system deployment in lower environments, ideally before they appear in production. For this test study, the focus of the testing approach was system performance and end-user experience.

Each component was monitored as the workflows were conducted against different load scenarios. Upon test completion, results were assembled and analyzed to identify both bottlenecks and over-resourced components in the system. This information was used to identify system components that needed to be scaled up, down, or out before further testing was repeated.

Manual user experience testing was conducted by capturing screen recordings of the workflow testers to ensure users of the system could complete their workflows productively.

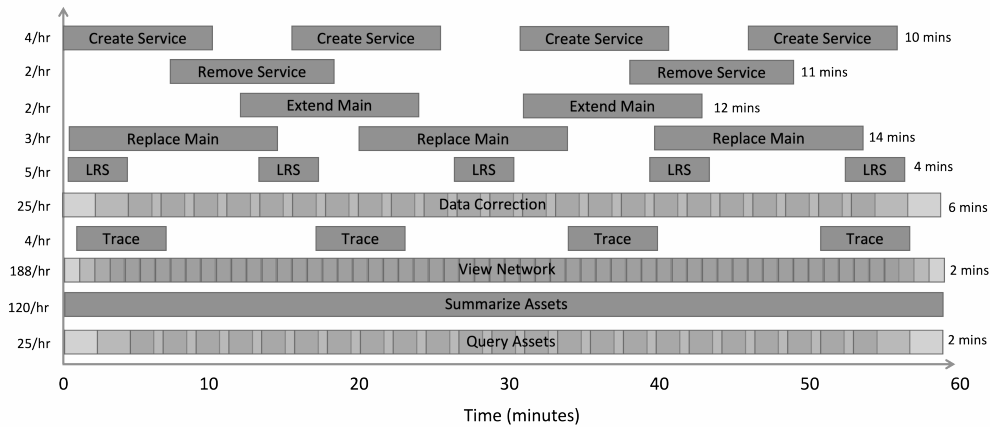
For more information, see how to [design an effective test strategy](#).

Workflow pacing

This test study applied a pacing model to the [tested workflows](#). The pacing model shows how the test intends to simulate the pace of work at a utility, where workflows are performed as some number of operations per hour across a team of staff resources. This approach was based on Esri customer input and aimed to match the small to medium gas utility customer scenario that the data was based on.

The various workflows were spread out through a one-hour test period and staggered so as to not start at the same time, while overlapping with each other as real-world workflows also would. This overall breakdown of workflow pacing is considered the “design load” that the system is subjected to. The load was then increased by multiplying the workflows to a point where the system was no longer able to provide acceptable responses or support successful workflows. Note that the workflow pacing model applied in this test study might not match typical daily use at your organization.

Performance testing tools



Performance testing tools

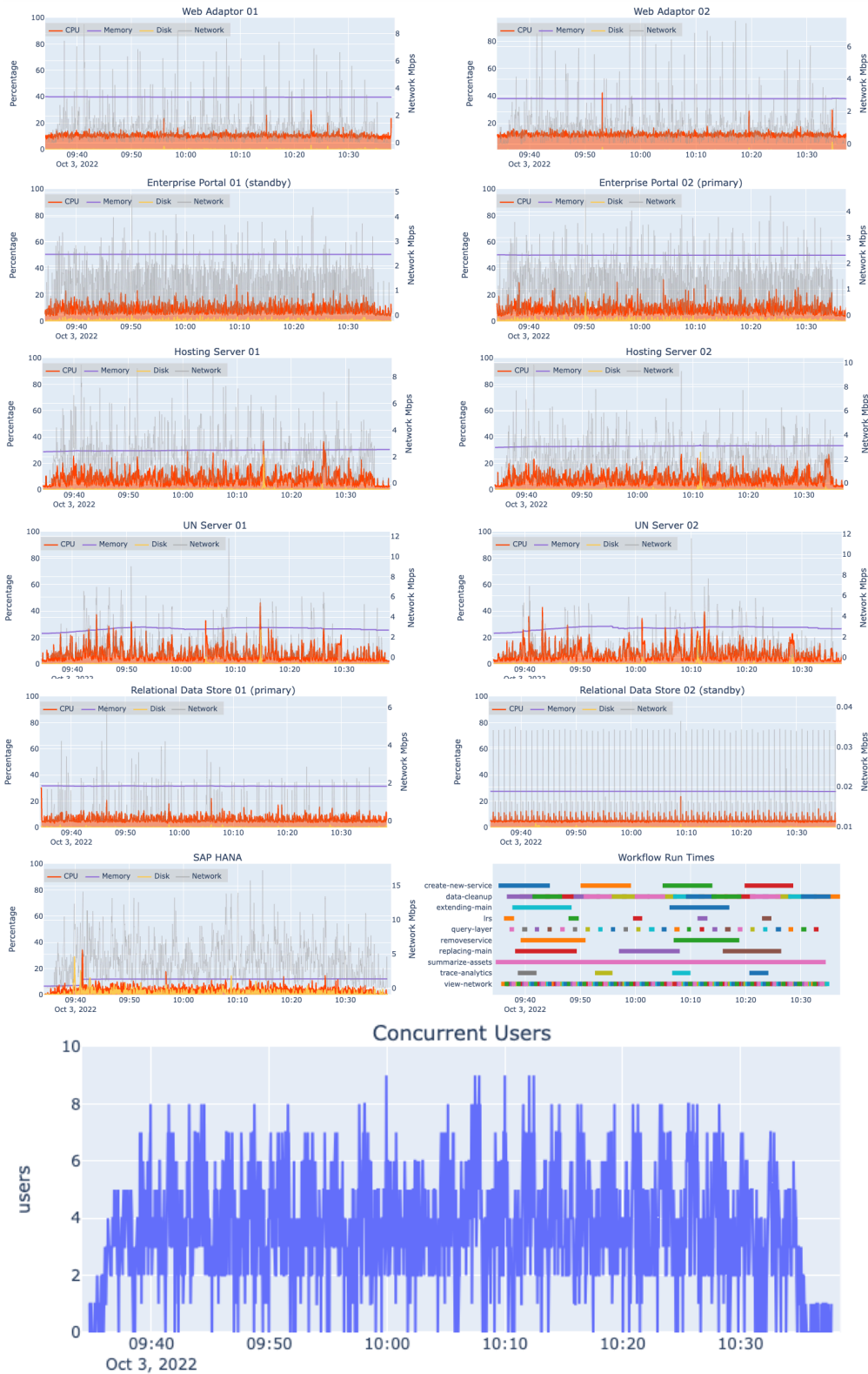
Because ArcGIS is a multi-tier system, performance tests were conducted across client, service, and data storage tiers, as well as the underlying infrastructure itself. In this test study, [JMeter](#) was used to simulate the user workflows and measure system performance under different loads. ArcGIS Pro requests were recorded and then replayed to simulate load in addition to manual workflows that were performed to assess end-user experience. Windows Performance Monitor and ArcGIS Monitor were also used to monitor resource utilization across different components.

For more information, see [tools for performance testing](#).

Test results

This architecture was validated with automated load tests and manual users in three scenarios, and you can see the results from each below. At a high level, the test results show that as implemented, the system is adequately resourced to support loads from the design load through 4x the design load. Tests also reinforced the importance of proper application and system configuration for performance. Across each scenario system utilization increases proportionally with load.

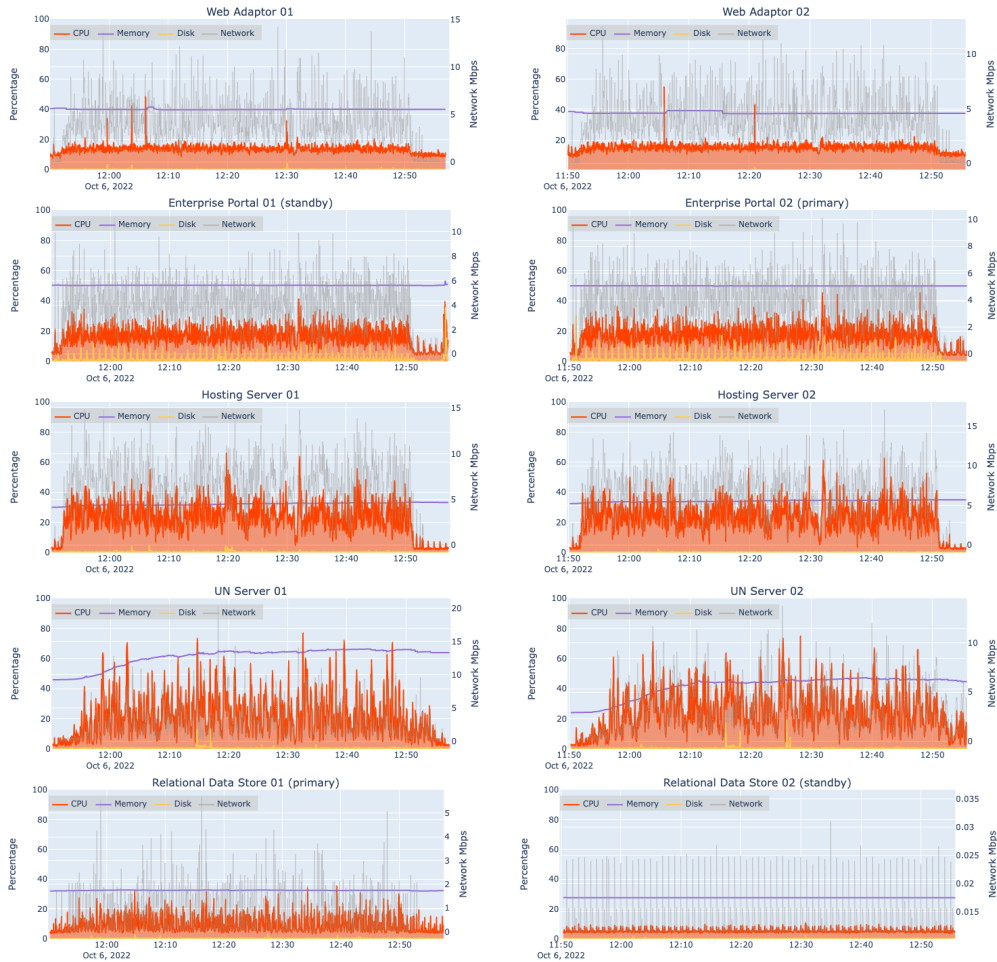
Test scenario: design load



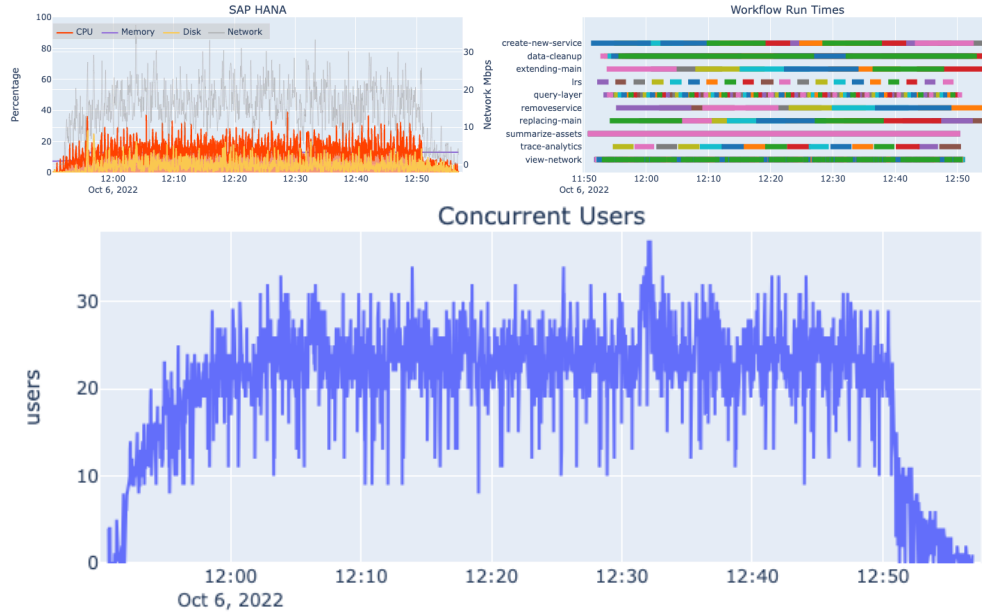
Test results

- The system supported the load
- The hosting servers generally ran below 20% CPU
- The GIS Servers generally ran below 20% CPU, with peaks around 40-60%
- The SAP HANA Server generally ran below 20% CPU

Test scenario: 4x design load

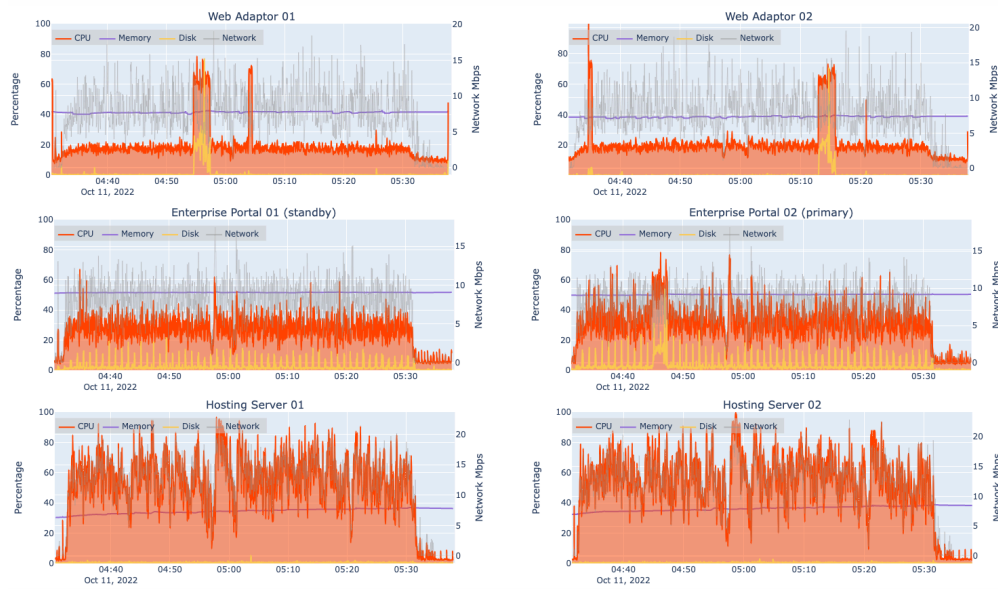


Test results

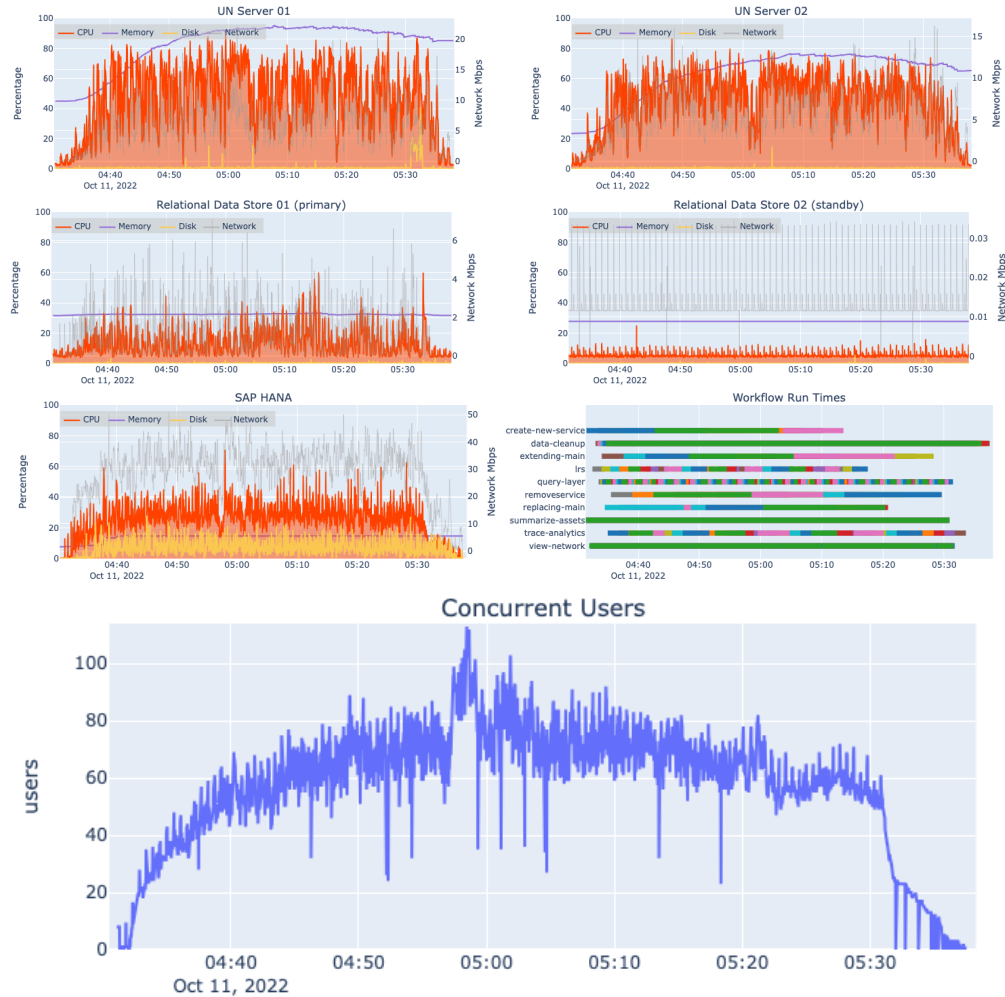


- The system supported the load
- The hosting servers generally ran above 40% CPU
- The GIS Servers generally ran above 60% CPU
- The SAP HANA generally peaked around 30% CPU

Test scenario: 8x design load



Test results

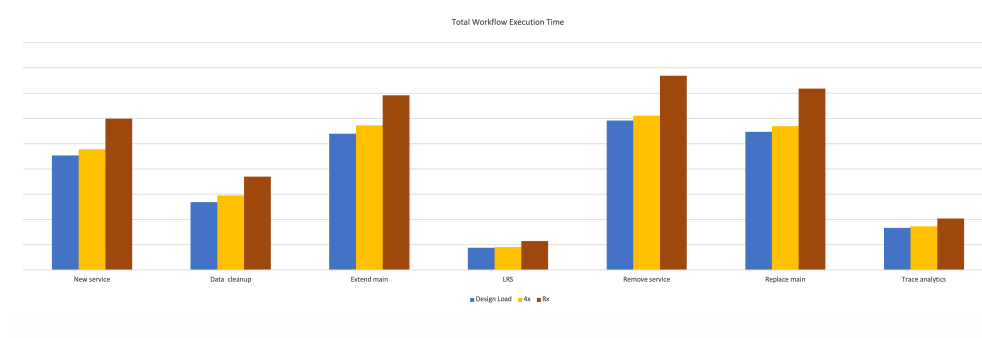


- The system did not support the load
- The hosting servers reached 100% CPU
- The GIS servers reached 90% CPU
- The SAP HANA reached 40-60% CPU
- The concurrent users ramped up excessively because requests are not being completed
- The system reported an excessive number of errors

User experience - conducted workflow times

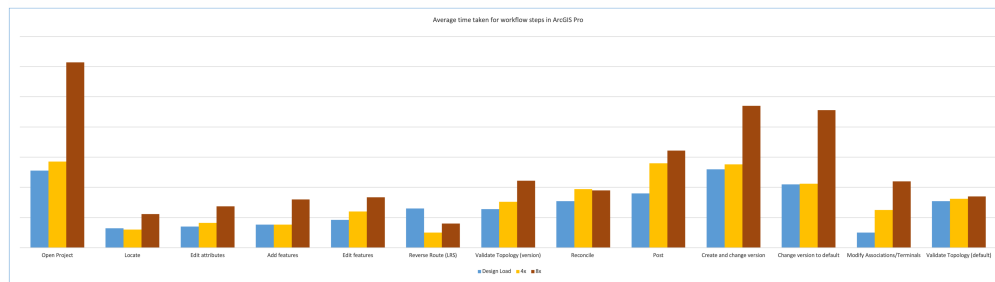
While the system was under load, conducted workflow times were captured as experienced by the users. This represents the time it took to complete all the steps listed in the [workflows](#). Conducted workflow times are consistent until the system becomes overloaded at 8x design load.

Test results



User experience - conducted workflow step times

While the system was under load, conducted workflow times of key steps across all workflows were captured. This represents the average time it took to complete a given step. Conducted workflow times are consistent until the system becomes overloaded at 8x design load.



How to use this test study

This test study has been prepared and presented as part of the ArcGIS Architecture Center to help organizations understand how a specific [reference architecture](#) can be designed and deployed on specific physical infrastructure, and how it may perform when tested under load. To make best use of this test study, the following approach is suggested:

1. **Understand** the [reference architecture](#) first. It contains important information and resources not duplicated in the test study.
2. **Assess** the workflows and data used in the test study against those your system supports. The system may support additional or different workloads, but making adjustments to the system, the workflows, and/or the load introduces risk.
3. **Use** the test study as your guide for implementing your Network Management System. Where your workflows and/or requirements differ, we offer a [design process](#) you can follow.
4. **Determine** possible improvement areas for your current implementation by considering design choices made in this test study.
5. **Evaluate** the testing approach used against your own and look for improvement areas.
6. **Learn** more about the architecture practices illustrated in the reference architecture in the [ArcGIS Architecture Center](#).

Additional related resources

These resources reference key concepts and practices that may not have been explicitly stated within the test study, but can be significant factors in designing, implementing, and operating a successful Network Management System.

- [Build and manage a GIS program](#)
- [IT Governance](#)
- [Upgrades and patching](#)
- [Backups and disaster recovery](#)
- [Secure network design](#)